Neural Computing and Applications (2021) 33:12661–12687
https://doi.org/10.1007/s00521-021-05915-w

ORIGINAL ARTICLE

Multi-objective search group algorithm for thermo-economic optimization of flat-plate solar collector

Bao-Huy Truong[1] · Perumal Nallagownden[2] · Khoa Hoang Truong[2] · Ramani Kannan[2] · Dieu Ngoc Vo[3,4] · Nguyen Ho[5]

Received: 19 May 2020 / Accepted: 9 March 2021 / Published online: 2 April 2021

Abstract

This study aims to develop a multi-objective version of the search group algorithm (SGA) called the multi-objective search group algorithm (MOSGA) to help determine thermo-economic optimization of flat-plate solar collector (FPSC) systems. Search mechanisms of the SGA were modified to determine non-dominated solutions through mutation, generation, and selection stages. Authors also mined the Pareto archive with a selection mechanism to maintain and intensify convergence and distribution of solutions. The study tested the proposed MOSGA with well-known multi-objective benchmark problems. Results were compared with outcomes from conventional algorithms using the same performance metrics to validate the capability and performance of the MOSGA. Afterward, MOSGA was applied to find the best design parameters to simultaneously optimize thermal efficiency and the total annual cost of FPSC systems. Four case studies were conducted with four different working fluids (pure water, SiO_2, Al_2O_3, and CuO nanofluids). Optimization results obtained by the MOSGA were analyzed and compared with solutions provided by other algorithms. The findings revealed relative improvement in thermal efficiency and reduced annual cost for all nanofluids compared to pure water. Thermal efficiency was improved by 2.2748%, 2.4298%, and 2.7948% for SiO_2, Al_2O_3, and CuO case studies, respectively, compared to pure water. Meanwhile, TAC rates were increased by 2.4111%, 2.3403%, and 2.9133% for these case studies, respectively. Comparative results also demonstrated that MOGSA was robustly effective and superior in the selection of appropriate design parameters of FPSC systems.

Keywords Search group algorithm · Multi-objective optimization · Flat-plate solar collector · Thermo-economic optimization

✉ Dieu Ngoc Vo
vndieu@hcmut.edu.vn

1 Institute of Engineering and Technology, Thu Dau Mot University, Thu Dau Mot City, Binh Duong Province, Vietnam

2 Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia

3 Department of Power Systems, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam

4 Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam

5 FPT University, Hanoi, Vietnam

1 Introduction

1.1 Background

The heat consumption for the domestic and industrial sectors occupies a large portion of the total energy consumption [1]. Most thermal systems rely on fossil fuel combustion; however, overuse of conventional fuels (oil, coal, natural gas) detrimentally affects the environment through harmful emissions that exacerbate global warming [2, 3]. Renewable energy having sustainability and eco-friendliness provides a perfect solution for limiting fossil fuel consumption and environmental issues. In particular, solar energy has enormous potential for heat production. Solar thermal systems can fit with demands for low- and medium-temperatures that account for more than 50% of the industrial heat demand [4, 5]. The solar collector is a critical component of solar thermal systems to transform solar radiation into heat energy. The flat-plate solar collector (FPSC) dominates the market for low and medium thermal applications, especially for temperatures below 100 °C [6]. Two key obstacles to solar thermal system development are low thermal efficiency and high operational cost of the solar collector [7]. Hence, the operation of the FPSC system with maximized efficiency and minimized costs is a vital developmental objective.

1.2 Literature review

1.2.1 Optimization of FPSC system

Over the last decade, various studies explored optimization approaches for the FPSC system and gained impressive results. Search group algorithm (SGA) [8] was recently suggested for the energetic optimization of a solar water heating (SWH) system using an FPSC. As a result, energy efficiency after the optimization process was increased by 4.904% compared to the base case. Farahat et al. [9] developed exergetic optimization by applying sequential quadratic programming (SQP) to optimize FPSC efficiency by minimizing exergy losses. Jafarkazemi et al. [10] introduced a model for energetic and exergetic evaluation of FPSC to analyze the impacts of all design variables on efficiency. Badr et al. [11] exploited a genetic algorithm (GA) to optimize an active SWH with FPSC under different environmental conditions and design parameters. Wenceslas [12] optimized a thermosyphon solar water heater using GA and optimal results of design parameters to fabricate an FPSC with locally available materials. This system obtained higher efficiency with lower collector surface area. Khademi et al. [13] compared SQP with GA to maximize exergy performance of FPSC. They found that the optimization results of the GA method yielded higher accuracy but lower convergence speed than SQP. Several studies recently conducted meta-heuristic algorithms to investigate the efficiency of a smooth flat-plate solar air heater (SFPSAH). Siddhartha et al. [14] studied thermal performance optimization for SFPSAH using particle swarm optimization (PSO). Their results indicated improved efficiency after raising the number of glass cover and heat transfer rates. The optimal performance came out at 72.42%. Siddhartha and Chauhan [15] carried out a theoretical study using simulated annealing (SA) to predict optimal points for operating parameters to increase SFPSAH efficiency. The GA technique [16] and a stochastic iterative perturbation method [17] were applied to maximize the efficiency of SFPSAH. In general, by raising the Reynolds number as well as tilt angle and number of covers, efficiency was enhanced in all case studies. Rao et al. [18] used theoretical analysis to determine design parameters and performance of SFPSAH by employing a teaching learning-based optimization (TLBO) method. Results indicated that TLBO provided good flexibility and convergence speed compared to GA and PSO. Different methods, namely the artificial bee colony algorithm (ABC) and GA, were developed by Sahin [19] to examine the correlations between different parameters in SFPSAH. Results showed that ABC improved efficiency slightly more than GA. Yildirim [20] implemented a study to analyze the thermohydraulic condition of single-pass solar air heaters by optimizing channel depth and airflow rate using the ABC algorithm. Jiandong [21] analyzed numerical simulations of configuration parameters and their impacts on FPSC performance.

Recently, several studies were conducted to explore the economic aspects of FPSC. Bornatico [22] proposed the PSO method to evaluate optimized values for key components of a solar thermal system to minimize installation costs and energy consumption for an entire building. The genetic algorithm (GA) was implemented to estimate the maximal life-cycle savings of FPSC for 182 plants in Chile [23]. Outcomes pointed out that optimized operations for FPSC systems benefited in all 182 locations. A theoretical study carried out GA to design a solar water heater on a life-cycle cost basis [24]. Several studies applied hybrid techniques to optimize problems confronting the life-cycle costs of FPSC, including the Hooke-Jeeves method with PSO [25, 26] and the binary search method with GA [27].

Despite impressive results, all models cited focused on solving single-objective problems to maximize FPSC efficiency or minimize costs separately. Recent studies have looked over the multi-objective optimization of FPSC. Hajabdollahi [28] analyzed multi-objective particle swarm optimization (MOPSO) to optimize cost and efficiency simultaneously. Outcomes indicated improved thermal-economic value at a lower rate of heat transfer. Hajabdollahi and Premnath [29] also applied MOPSO to minimize yearly cost and maximize efficiency by analyzing effects from Al_2O_3 nanoparticles and various design parameters of FPSC. They found that the addition of Al_2O_3 nanoparticles improved performance by 2% and decreased cost by 3.5%. Similar work was pursued by Hajabdollahi [30], in which an FPSC system using CuO nanofluid was modeled and optimized using MOPSO. This nanofluid notably enhanced efficiency and decreased cost compared to the pure fluid. Hajabdollahi et al. [31] investigated effects on FPSC systems from SiO_2, Al_2O_3, and CuO nanofluids in thermo-economic terms. Cost and efficiency were determined using a non-dominated sorting genetic algorithm II (NSGA-II). Results showed that all nanofluids improved efficiency and reduced TAC.

From the above literature, a vast majority of studies [8–27] focused on optimizing the FPSC system with a single objective function to maximize FPSC efficiency or minimize costs separately. Very few studies [28–31] optimized the FPSC system considering multi-objective functions for both efficiency and cost. A practical optimization problem often contains more than one objective to be optimized. These motivate the researchers to formulate a multi-objective optimization problem to find a set of trade-off solutions in the search domain. Therefore, FPSC should be concurrently optimized in terms of the thermo-economic viewpoints to assist manufacturers by offering trade-offs between thermal efficiency and cost.

1.2.2 Search group algorithm

The search group algorithm (SGA) is a recently developed metaheuristic algorithm by Gonçalves et al. [32]. SGA mechanism is based on creating and developing search groups based on the promising individuals obtained, which aims to generate an appropriate balance between two capabilities of optimization (exploration and exploitation). The exploration process shows the algorithm's ability to find promising regions on the design domain, i.e., the regions where the optimal solution can be found. Meanwhile, the exploitation process demonstrates the algorithm's ability to refine the solution on these promising regions, i.e., to perform a local search on them. Both processes are vital to achieving an optimal solution. SGA is a search method with superiority exploration and exploitation; hence, it has achieved promising results when solving different engineering optimization problems.

In [32], the authors proposed SGA to optimize the truss structures. The simulation results showed that SGA obtained the lightest structures for five out of six case studies in comparison with other methods such as Finite Element Force, GA, PSO methods, and a hybrid optimality criterion and GA. Pedro et al. [33] proposed SGA to design of steel–concrete composite I-girder bridges. Statistical analysis showed that SGA had the best performance out of four well-known optimization methods, including the GA method, firefly algorithm (FA), backtracking search algorithm (BSA), and imperialist competitive algorithm (ICA). For the discrete optimization problem, Carraro et al. [34] applied SGA to deal with the optimization of three planar steel frame designs. The authors pointed out that SGA had effective heuristic mechanisms, which help avoid getting stuck in local optimization. As a result, SGA had better performance state-of-the-art methods. Noorbin and Alfi [35] developed a fuzzy SGA (FSGA) method to improve the SGA solution quality. The FSGA was used to adjust the controller parameters for the network-based control system. From the simulation results, the FSGA proved its feasibility in the field of control systems. Khamari et al. [36, 37] proposed SGA with a PID controller for an application in automatic generation control (AGC). The overall performance of the SGA-based PID controller was very effective, which obtained better performance compared with the Firefly Algorithm-based PID controller. Acampora et al. [38] suggested SGA to deal with optimal reactive power flow on IEEE 57-bus and 118-bus systems. The performance of SGA statistically outperformed other algorithms at a 90% confidence level. A review of the literature undertaken found that SGA is a competitive metaheuristic algorithm for engineering design applications. Since SGA is relatively new and promising, SGA is potential to be further studied and exploited to effectively solve multi-objective problems by integrating with appropriate mechanisms.

1.3 Motivation and aim

For the problems considered in this paper, studies on the multi-objective optimization of FPSC are lacking in the literature. FPSC optimization problems with multiple objectives need to be further studied deeply. Previous studies of this area concentrated on implementing conventional algorithms, namely MOPSO and NSGA-II, without taking into account other recent methods. Moreover, when solving this problem, comparisons of solution quality have not yet been considered. Hence, it is high time that efforts are dedicated to applying new multi-objective algorithms to solve this problem more effectively.

Recently, many metaheuristic algorithms have been continuously developed to solve multi-objective problems such as multi-objective water cycle algorithm (MOWCA) [39, 40], multi-objective grey wolf optimizer (MOGWO) [41], multi-objective symbiotic organisms search (MOSOS) [42], multi-objective multi-verse optimizer (MOMVO) [43], and multi-objective lightning attachment procedure optimization (MOLAPO) [44], to name just a few. However, the no-free-lunch (NFL) theorem [45] logically proved that no metaheuristic algorithm could solve all optimization problems efficiently. For the algorithm proposed in this paper, the main advantage of SGA is to generate a proper balance between exploration and exploitation so that it can compete with other metaheuristic algorithms in terms of performance and robustness. Furthermore, the simulation results of SGA verified that it was appropriate and competitive for solving different engineering problems such as truss structure optimization [32, 33], optimization of planar steel frames [34], networked control systems [35], automatic generation control [36, 37], and optimal voltage regulation in power systems [38]. To our best knowledge, SGA has not been developed to deal with multi-objective problems, especially multi-objective optimization of FPSC systems.

Therefore, with the above motivations, this study proposed a new multi-objective search group algorithm (MOSGA) for thermo-economic optimization of FPSC (TEO-FPSC) problem. The proposed MOSGA is the first multi-objective optimization version of the original SGA technique, which is a significant contribution of this study. Elitist non-dominated sorting technique and Pareto archive was integrated into SGA search mechanism to develop new MOSGA. It was developed to achieve fast convergence and maintain diverse solutions in the non-dominated set. The TEO-FPSC problem was formulated with two objective functions: thermal efficiency and total annual cost (TAC). Specification parameters of the FPSC system, including mass flow rate, riser tube outer diameter, tube number, insulation thickness, and nanoparticle concentration, were selected as design variables. MOSGA was developed to provide Pareto optimal set and respective trade-offs for two contradictory objectives, namely maximized thermal efficiency and minimized TAC, without compromising each objective. Moreover, the decision-making approach was applied to determine the best compromise solution.

1.4 Contributions

The contributions of this paper are outlined as follows:

1. A new multi-objective algorithm (MOSGA) was developed to solve the TEO-FPSC problem, where thermal efficiency and total annual cost were simultaneously optimized.
2. The proposed MOSGA was validated on eight benchmark multi-objective problems with diverse features and compared results with well-regarded multi-objective optimization techniques. The comparative results showed that Pareto optimal solutions obtained by MOSGA provided better convergence and distribution than other techniques.
3. The MOSGA was implemented to simultaneously optimize thermal efficiency and the total annual cost of FPSC systems under steady-state conditions. Four case studies are considered with four different working fluids (pure water, SiO_2, Al_2O_3, and CuO nanofluids).
4. Statistical comparisons and analyses of optimized results demonstrated the effectiveness of the MOSGA, offering robust solutions for the TEO-FPSC problem.

1.5 Paper outline

Section 2 introduces the problem formulation for the TEO-FPSC problem. Section 3 describes the proposed MOSGA, along with performance metrics used for the comparison of multi-objective algorithms. The decision-making method is also given in this section. Section 4 begins illustrating and discussing the results of the MOSGA for multi-objective benchmark problems and is followed by optimization results for the TEO-FPSC problem. Finally, Sect. 5 makes a conclusion for this paper.

2 Problem formulation

2.1 Thermo-economic modeling of FPSC

Figure 1 depicts a typical FPSC. This section presents thermo-economic modeling of FPSC for water heating systems under steady-state conditions.

2.1.1 Thermal efficiency

The first objective function for thermal efficiency of FPSC can be proposed as:

$$\eta = \frac{Q_{\mathrm{u}}}{A_{\mathrm{c}} I_{\mathrm{T}}} \tag{1}$$

where Q_{u} is useful heat gain, A_{c} is cover surface area, and I_{T} is total solar radiation intensity.

Useful heat gain of FPSC is calculated as follows [46]:

$$Q_{\mathrm{u}} = A_{\mathrm{p}}(\tau\alpha)I_{\mathrm{T}} - A_{\mathrm{c}}U_{\mathrm{L}}\left(T_{\mathrm{pm}} - T_{\mathrm{a}}\right) \tag{2}$$

where A_{p} is absorber plate area, U_{L} is the overall heat loss coefficient, T_{pm} is the mean temperature of absorber plate, T_{a} is ambient temperature, and $(\tau\alpha)$ is effective transmittance-absorptance.

The overall heat loss coefficient (U_{L}) is the sum of the top, edge, and back loss coefficients:

$$U_{\mathrm{L}} = U_{\mathrm{t}} + U_{\mathrm{e}} + U_{\mathrm{b}} \tag{3}$$

The top loss coefficient is obtained by Klein's empirical formula [47]:

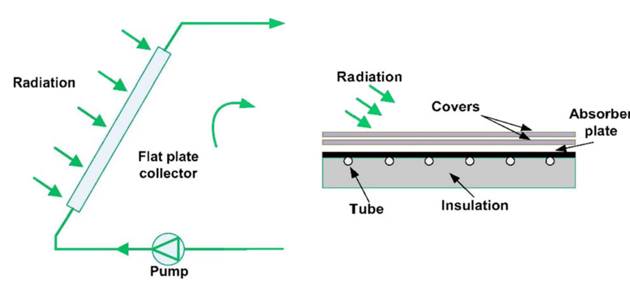


Fig. 1 Diagram of a flat-plate solar collector [28]

$$U_t = \left[\frac{N}{\frac{C}{T_{pm}}\left[\frac{T_{pm}-T_a}{N-f}\right]^e} + \frac{1}{h_w} \right]^{-1} + \frac{\sigma(T_{pm}-T_a)\left(T_{pm}^2+T_a^2\right)}{(\varepsilon_p+0.00591Nh_w)^{-1}+\frac{2N+f+1+0.133\varepsilon_p}{\varepsilon_g}-N} \tag{4}$$

For Eq. [4], f, C, e, and h_w are defined as follows:

$$f = (1 + 0.089h_w - 0.1166h_w\varepsilon_g)(1 + 0.07866N) \tag{5}$$

$$C = 520(1 - 0.000051\beta^2), \quad \begin{cases} 0 < \beta < 70^\circ \\ \beta = 70^\circ \text{ if } \beta > 70^\circ \end{cases} \tag{6}$$

$$e = 0.430 \times \left(1 - \frac{100}{T_p}\right) \tag{7}$$

$$h_w = 5.7 + 3.8v \tag{8}$$

where N is the number of glass cover, h_w is the heat transfer coefficient of the wind, v is the wind speed, ε_g is the emissivity of the glass cover, ε_p is the emissivity of the absorber plate, σ is the Stefan–Boltzmann constant, and β is the slope of the collector.

Edge and back loss coefficients are calculated as follows:

$$U_e = \frac{k_e}{\delta_e} \times \frac{A_e}{A_c} \tag{9}$$

$$U_b = \frac{k_b}{\delta_b} \tag{10}$$

where A_e is the heat transfer surface area of the edge, k_e and δ_e are the thermal conductivity and insulation thickness of the edge, respectively, k_b and δ_b are the thermal conductivity and insulation thickness of the back, respectively.

Mean temperature (T_{pm}) is estimated by assuming an initial value to estimate U_L and Q_u. The next value of T_{pm} is then calculated as [48]:

$$T_{pm} = T_i + \frac{Q_u}{A_p F_R U_L}(1 - F_R) \tag{11}$$

where T_i is the fluid inlet temperature; and the heat removal factor (F_R) can be determined as:

$$F_R = \frac{\dot{m}C_p}{A_p U_L}\left[1 - \exp\left(-\frac{F' U_L A_p}{\dot{m}C_p}\right)\right] \tag{12}$$

$$F' = \frac{\frac{1}{U_L}}{W\left(\frac{1}{U_L[D_o+(W-D_o)F]} + \frac{1}{C_b} + \frac{1}{\pi D_i h_{fi}}\right)} \tag{13}$$

$$F = \frac{\tanh\left[\frac{m(W-D_o)}{2}\right]}{\frac{m(W-D_o)}{2}} \tag{14}$$

$$m = \sqrt{\frac{U_L}{k\delta}} \tag{15}$$

where $\dot{m}$ is mass flow rate, D_o and D_i are the outer and inner diameter of the riser tube, respectively, W is tube spacing, C_b is the thermal conductance of the bond, h_{fi} is the convection heat transfer coefficient between the fluid and tube wall, k and δ are thermal conductivity and thickness of the absorber plate, respectively.

The properties of nanofluid are estimated by employing regression equations as follows [49–52]:

$$\rho_{nf} = \phi\rho_{np} + (1 - \phi)\rho_{bf} \tag{16}$$

$$C_{p,nf} = \frac{\phi(C_p)_{np} + (1-\phi)(C_p)_{bf}}{\rho_{nf}} \tag{17}$$

$$\frac{k_{nf}}{k_{bf}} = \frac{k_{np} + 2k_{bf} - 2\phi(k_{bf} - k_{np})}{k_{np} + 2k_{bf} + \phi(k_{bf} - k_{np})} + \frac{\phi(\rho C_p)_{np}}{2k_{bf}}\sqrt{\frac{2\kappa_B T}{3\pi d_{np}\mu_{bf}}} \tag{18}$$

$$\frac{\mu_{nf}}{\mu_{bf}} = 1 + 7.3\phi + 123\phi^2 \tag{19}$$

where ϕ is particle concentration, κ_B is Boltzmann constant, d_{np} is the nanoparticle size, subscripts nf, bf, and np are nanofluid, base fluid, and nanoparticle, respectively.

2.1.2 Economic analysis

The second objective function for Total Annual Cost (TAC) of FPSC can be expressed as [31]:

$$C_{total} = aC_{inv} + C_{op} \tag{20}$$

Investment cost (C_{inv}) can be expressed according to Hall's correlation method [53]:

$$C_{inv} = \varphi\left\{a_1(A_p)^{b_1} + a_2(A_{tube})^{b_2} + a_3(\forall_{insu})^{b_3} + a_4(A_c)^{b_4}\right\} + a_5(\dot{W}_p)^{b_5} + k_{np}L_t N_t \phi\rho_{np}\left(\frac{\pi D_i^2}{4}\right) \tag{21}$$

where φ is the collector assembly coefficient, A_{tube} is the outside surface area of the tube, $\forall_{insu}$ is insulator volume, $\dot{W}_p$ is pump power, m_{np} is nanoparticle mass, k_{np} is the unit price of the nanoparticle, L_t is the length of the tube, and N_t is the number of riser tubes.

Annual cost factor (a) is calculated as:

$$a = \frac{i}{1 - (1 + i)^{-y}} \tag{22}$$

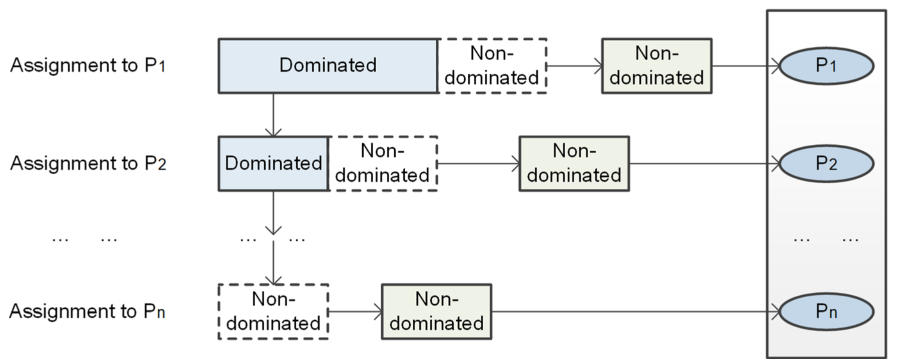
in which y is the lifetime of the system, and i is the inflation rate.

Operational cost is calculated as follows:

$$C_{op} = N_h k_{el} \dot{W}_p \tag{23}$$

where N_h is the system's operational hours per year, and k_{el} is the unit value of electricity.

Fig. 2 Schematic view of fast non-dominated sorting technique



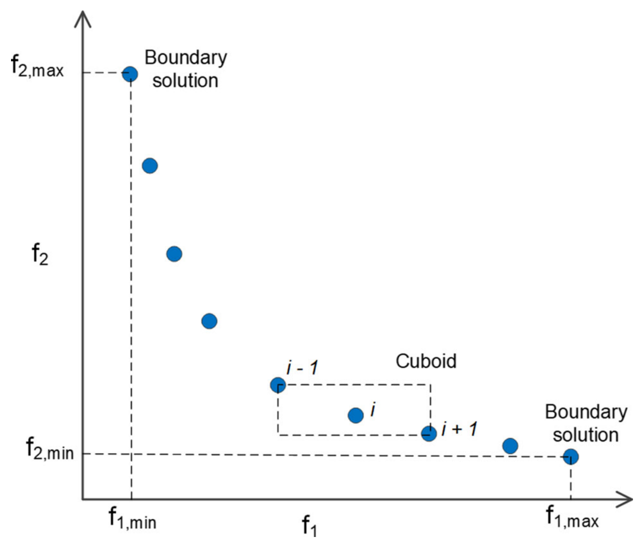


Fig. 3 Schematic view of crowding distance computation

2.1.3 Thermal modeling process

An iterative process was implemented to compute thermal efficiency and TAC for the FPSC system as follows:

Step 1: At the beginning of the iteration, the absorber plate's mean temperature (T_{pm}) is assumed based on the fluid's inlet temperature ($T_{pm} = T_i + 10$);
Step 2: Top loss (U_t), edge loss (U_e), back loss (U_b), and consequent overall loss coefficients (U_L) are calculated according to Eqs. (3–10);
Step 3: By applying the overall heat loss coefficient, both the heat removal factor (F_R) and useful energy output (Q_u) are computed by using Eqs. (12) and (2), respectively;
Step 4: The new mean temperature of the absorber plate is adjusted using Eq. (11);
Step 5: This new T_{pm} is compared to the previous value. If the difference is within the acceptable boundary, the process stops and moves on to Step 6; if it exceeds this limit, the new T_{pm} is adopted as a replaced value, and the process repeats from Step 2;
Step 6: When a correct value for T_{pm} is obtained, efficiency and TAC are defined by Eqs. (1) and (20), respectively;

2.2 Formulation for TEO-FPSC problem

In this research, both thermal efficiency and TAC of FPSC system were considered as objective functions for simultaneous optimization. Hence, thermo-economic optimization of FPSC (TEO-FPSC) problem is defined as follows:

$$\text{Find: } x^* = [\dot{m}, D_o, N_t, \delta_b, \phi] \tag{24}$$

$$\text{Maximize: } \eta(x^*) \tag{25}$$

$$\text{Minimize: } C_{total}(x^*) \tag{26}$$

$$\text{Subject to: } 0.01 \le \dot{m} \le 0.1 \tag{27}$$

$$0.005 \le D_o \le 0.015 \tag{28}$$

$$6 \le N_t \le 20 \tag{29}$$

$$0.02 \le \delta_b \le 0.1 \tag{30}$$

$$0.001 \le \phi \le 0.1 \tag{31}$$

where mass flow rate ($\dot{m}$), outer diameter of the riser tube (D_o), tube number (N_t), insulation thickness (δ_b), and nanoparticle concentration (ϕ) are design variables in the optimization procedure.

3 Multi-objective search group algorithm

The search group algorithm (SGA) is a population-oriented metaheuristic algorithm [33]. To obtain feasibly optimized solutions, SGA creates a search group to explore promising regions in its global search mode then exploits the best design from promising domains in local search [34]. Its main advantage is the balance between exploration and exploitation of the search space [32]. The present work

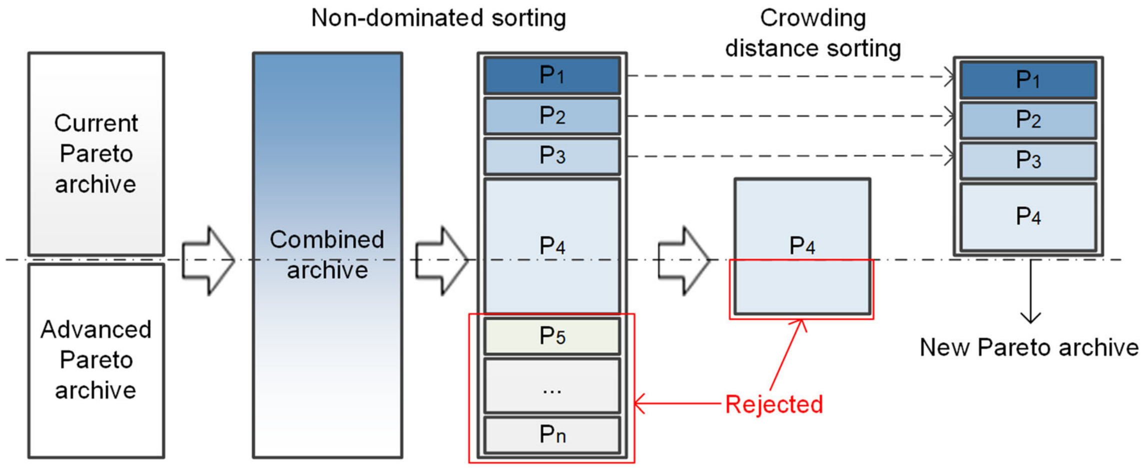


Fig. 4 Pareto archive selection mechanism

proposed a modified SGA approach to solve multi-objective problems effectively. Our proposed search mechanism was inspired by the original SGA in which solutions were obtained through mutation, generation, and selection stages. To develop this protocol, elitist non-dominated sorting technique and Pareto archive selection were integrated to produce MOSGA. These new techniques and the new algorithm's process are now described.

3.1 Elitist non-dominated sorting technique

The elitist non-dominated sorting technique [54] consists of two techniques: fast non-dominated sorting approach and crowding distance computation, used by the proposed MOSGA to sort a population into different non-dominated fronts with computed crowding distance.

The fast non-dominated sorting approach is firstly adopted to identify different non-dominated fronts in a population. Firstly, two entities are defined for each solution of the population, in which domination count n_i is the number of solutions that dominate the solution i, and S_i is a set of solutions that is dominated by solution i. All solutions with a domination count n_i of zero are placed on the first non-dominated front. Secondly, for every solution i with $n_i = 0$, it visits each solution j in the set S_j and reduces its domination count n_j by one. If any solution j has a domination count n_j of zero, then it is placed on the second non-dominated front (a separate list J). Then, the above process continues with each solution of the second non-dominated front to identify the third non-dominated front. This process is continuously carried out until all non-dominated fronts are obtained. Figure 2 presents a schematic view of the fast non-dominated sorting technique.

Crowding-distance computation is then implemented for diversity preservation of non-dominated solutions in a particular front. This parameter shows the density of solutions surrounding a particular solution in the population. First, the population is sorted in ascending order of the magnitude of each objective value. The boundary solutions for each objective function (solutions with the minimum and maximum objective values as depicted in Fig. 3) are assigned an infinite distance value. All other intermediate solutions are assigned a crowding distance value as follows:

$$d_j^i = \sum_{j=1}^{m} \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{\max} - f_j^{\min}} \tag{32}$$

where m is the number of objective functions; f_j^{i+1} and f_j^{i-1} denote the jth objective function values for two adjacent solutions ($i+1$ and $i-1$) of solution i, respectively, $f_j^{\max}$ and $f_j^{\min}$ designate the maximum and minimum values of the jth objective function, respectively.

A solution with a higher crowding distance value indicates that it is located in a lesser crowded region by other solutions. The MOSGA applied the crowded-comparison operator ($\prec_n$) to select the better solution in multi-objective space based on two attributes of each solution: non-dominated rank (r) and crowding distance (d) as follows:

$$i \prec_n j \text{ if } \left(r_i < r_j\right) \text{ or } \left(\left(r_i = r_j\right) \text{ and } \left(d_i > d_j\right)\right) \tag{33}$$

This means that, if two solutions belong to different non-dominated ranks, the solution of the better non-dominated rank will be preferred. Meanwhile, in case both solutions have the same non-dominated rank, the solution with the higher crowding distance value will be preferred.

3.2 Pareto archive selection

A vital task of multi-objective optimization is to save non-dominated solutions in a Pareto archive. The archive is updated based on a selection mechanism after each iteration. The selection instrument helps to avoid the loss of potential candidate solutions as eliminating solutions. As mentioned, all newly created family members are saved in an advanced archive. After each iteration, the algorithm combines two archives (current and advanced), after which the total size of the set is larger than the limited size. The limitation of the Pareto archive is fixed by a selection mechanism that discards undesirable solutions.

First off, the combined Pareto archive is sorted via a fast non-domination sorting technique into different non-domination levels (P_1, P_2, …, P_n). Solutions of the best non-domination level (P_1) are the first to be selected to come into the new Pareto archive. If the P_1 size is smaller than the limited size of the archive, all P_1 members enter the new Pareto archive. Remaining solutions for the new Pareto archive are chosen from subsequent non-domination levels in ranking order (P_2, P_3 …). The process keeps up until the new archive has a sufficient number of levels for n_{pop} members—assuming that level P_k is the last non-domination level, beyond which, no other level can be accommodated. To select precise solutions for the new archive, solutions from the last front (P_k) are selected based on crowding distance value in descending order [54]. Figure 4 presents a schematic view of the Pareto archive selection procedure.

3.3 The proposed MOSGA

3.3.1 Population initialization

An initial population ($\mathbf{P}$) of n_{pop} individuals is created within the search space as follows:

$$P_{ij} = x_j^{\min} + \left(x_j^{\max} - x_j^{\min}\right) U[0,1] \quad \text{for } j = 1,\ldots,n,\ i = 1,\ldots,n_{pop}, \tag{34}$$

where P_{ij} represents the jth design variable of the ith individual of $\mathbf{P}$, U[0,1] is a stochastic variable between [0,1], $x_j^{\max}$ and $x_j^{\min}$ are upper and lower boundaries of the jth design variable, respectively, and n is the number of design variables.

3.3.2 Selecting initial search group

After initialization, objective functions for individuals in initial population $\mathbf{P}$ are evaluated. In single-objective optimization, individuals are ranked depending on objective function values so that the best individual is one with the best objective function value. However, in multi-objective optimization, individuals are ranked into different non-dominated fronts using the elitist non-dominated sorting technique. Afterward, n_g individuals are chosen from $\mathbf{P}$ to form a search group $\mathbf{R}$ based on its rank in the non-domination fronts. This step is done by using a standard tournament selection [55]. After each iteration, search group members are ranked to identify the best member for search group $\mathbf{R}$.

3.3.3 Mutation of search group

To enhance global searchability, n_{mut} members with a low ranking in non-domination fronts of search group $\mathbf{R}$ are selected for mutation by using inverse tournament selection. This approach creates new designs away from current

Table 1 Pseudocode of the proposed MOSGA

1: Set parameters for the proposed MOSGA;
2: Initial population $\mathbf{P}$ is randomly generated using Eq. (34);
3: Evaluate the values of multi-objective functions for each individual;
4: Sort initial population $\mathbf{P}$ into different non-domination fronts using elitist non-dominated sorting technique and store them in the Pareto archive;
5: Generate initial search group $\mathbf{R}^k$ choosing n_g solutions from $\mathbf{P}$ employing tournament selection;
6: Mutate n_{mut} individuals by newly generated members by Eq. (35);
7: Generate the families $\mathbf{F}_i$ using Eq. (36) and save them in the advanced Pareto archive;
8: Combine the current archive and the advanced archive;
9: Select best solutions for entry into the new Pareto archive using the Pareto archive selection mechanism;
10: Select a new search group as follows:
– Global phase: search group $\mathbf{R}^{k+1}$ is created by the best member of each family;
– Local phase: search group $\mathbf{R}^{k+1}$ is created by the best n_g solutions from Pareto archive
11: Update α^{k+1} using Eq. (37);
12: Set $k = k + 1$, if $k > it^{\max}$—move to Step 13; if otherwise back to Step 6;
13: Solution found: Pareto optimal solutions in final Pareto archive

members' locations so that new design space regions can be explored further. Mutation of new individual is performed per Eq. (35):

$$x_j^{\mathrm{mut}} = E\left[\mathrm{R}_{:,j}\right] + t\varepsilon\sigma\left[\mathrm{R}_{:,j}\right] \quad \text{for } j = 1, \ldots, n, \tag{35}$$

where x_j^{mut} represents the jth design variable of a mutated individual, E and σ are the mean value and standard deviation operators, $R_{:,j}$ denotes the jth column search group matrix, t is the distance adjustment value for a newly created individual, and ε is the convenient stochastic variable.

3.3.4 Formation of families

Each member of a search group is considered a family leader. A family is a set of family leader and individuals it creates. Each family leader creates a family as follows:

$$x_j^{\mathrm{new}} = R_{ij} + \alpha\varepsilon \quad \text{for } j = 1, \ldots, n, \tag{36}$$

where α adjusts the extent of the perturbation and is decreased for each iteration k as follows:

$$\alpha^{k+1} = b\alpha^k \tag{37}$$

where b is a parameter of the algorithm.

It should be noted that parameter α^k adjusts SGA to discover the design space. In initial SGA iterations, α^k must be a high value enough to allow family leaders to generate individuals spreading throughout the design space. This allows SGA to access new areas in a search domain in which a global solution can be found. As α^k decreases in value through SGA iterations, individuals created by family leaders tend to locate in its neighborhood. Moreover, the better the rank of a family leader in a search group is, the more individual members it creates. That is the family size for each leader based on its rank in the present search group. Moreover, all newly created individuals of the entire family are stored in the advanced Pareto archive for later sorting.

3.3.5 Selecting a new search group

Finally, selecting the new search group is a crucial stage having an impact on the convergence and diversity properties of the proposed algorithm. The MOSGA optimization process has two stages: global and local. In the global stage, all members of each family are sorted into the different non-domination front to determine the best family member. A member in the first non-dominated front with the best crowding distance value is considered as a new family leader, which is then used to create a new search group. This stage aims to explore most of the search space and diversify potential solutions. In the local stage, the selection mechanism is adjusted so that a new search group is created by choosing the best n_g members from the Pareto archive. Therefore, this stage exploits and refines the domain for the current best design.

The overall procedures of the proposed MOSGA are described in Table 1.

Main advantages of MOSGA to solve multi-objective problems are given as follows:

- The elitist non-dominated sorting technique was used as an appropriate method for identifying and sorting non-dominated solutions into different non-dominated ranks with calculated crowded distances. Hence, the MOSGA can effectively perform the next steps (mutation, generation, and selection) of the multi-objective optimization process.
- The mutation process is performed to continuously explore newer regions of the search space. It promotes exploration ability and avoids being stuck in the local front of the MOSGA concurrently during optimization.
- The perturbation parameter α^k is responsible for the adaptive transition from exploration to exploitation. Hence, the convergence of the MOSGA is assured.
- Better individuals create bigger families to obtain a better convergence for the process of optimization.
- Two proposed schemes to select the next search group (global and local stages) allow MOSGA to create an adequate balance between exploration and exploitation abilities.
- Tournament selection is utilized to select new search groups, which depicts a high probability of selecting individuals from less crowded regions. Based on this pattern, MOSGA found a diversity of solutions.
- The Pareto archive effectively stores the best non-dominated solutions obtained. Moreover, a selection mechanism is used to update this archive after each iteration to maintain the diversity of non-dominated solutions during optimization.

3.4 Performance metrics

Unlike single-objective optimization, Pareto optimal solutions in multi-objective optimization cannot be directly evaluated [56]. Therefore, there is a strong need for a set of performance metrics that can evaluate multi-objective algorithms properly. These metrics are now described below.

3.4.1 Generational distance

Van Veldhuizen et al. [57] proposed the Generational Distance (GD) metric to evaluate an algorithm's ability, generating a Pareto optimal front ($\mathrm{PF_g}$) that converges on the true Pareto optimal front (PF_{true}). The mathematical definition of this metric is:

$$\mathrm{GD} = \frac{\sqrt{\sum_{i=1}^{n_{pf}} d_i^2}}{n_{pf}} \tag{38}$$

where n_{pf} is the number of solutions in PF_g, d_i is the Euclidean distance between each solution in PF_g and the closest solution in PF_{true} in the objective space.

A multi-objective technique with the smallest GD value has the highest convergence on PF_{true}. This metric equals zero when all solutions of PF_g are on the PF_{true} curve.

3.4.2 Spacing

Scott [58] proposed the Spacing metric (SP) to evaluate the distribution of solutions in PF_g. This indicator estimates the relative distance between successive solutions as follows:

$$\mathrm{SP} = \sqrt{\frac{1}{n_{pf}-1}\sum_{i=1}^{n_{pf}}\left(d_i - \overline{d}\right)^2} \tag{39}$$

where $d_i = \min_{j\in\{1,2,\ldots,n_{pf}\}, i\neq j}\left(\left|f_1^i(x) - f_1^j(x)\right| - \left|f_2^i(x) - f_2^j(x)\right|\right)$, $i = 1, 2, \ldots, n_{pf}$ and $\overline{d}$ is the mean value of all d_i.

A multi-objective algorithm with a minimum SP metric has the best distribution in PF_g. This metric equals zero when all Pareto optimal solutions in PF_g are uniformly distributed.

3.4.3 Spread

Deb et al. [54] proposed the spread (Δ) metric to evaluate the extent of spread yielded by solutions in PF_g. This metric is valued by estimating the spread of extreme solutions as follows:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{n_{pf}}\left|d_i - \overline{d}\right|}{d_f + d_l + \left(n_{pf}-1\right)\overline{d}} \tag{40}$$

where d_f and d_l are the Euclidean distances between the extreme solutions in PF_{true} and PF_g, respectively; d_i is the Euclidean distance between neighboring solutions in PF_g, and $\overline{d}$ is the mean value of all d_i.

A lower Δ value implies better distribution and spread in PF_g. Hence, an algorithm with a minimal Δ metric obtains a better non-dominated set. For ideal distribution, Δ equals zero, indicating that true extremes of solutions have been identified and that the distribution of intermediate solutions is uniform.

3.4.4 Set coverage metric

Zeitzer [59] introduced the set coverage metric (*C*-metric) to evaluate the quality of solutions between two non-dominated sets, such that $C(X,Y)$, for example, estimates the percentage of solutions in Y that is weakly dominated by solutions in X [60]:

$$C(X,Y) = \frac{\left|\{y\in Y \mid \exists x\in X : x \preceq y\}\right|}{|Y|} \tag{41}$$

If $C(X,Y) = 1$, then all solutions in Y are weakly dominated by those in X or equal to solutions in X. If $C(X,Y) = 0$, then no solutions in Y are weakly dominated by those in X. Because the *C*-metric is not symmetric operator, both $C(X,Y)$ and $C(X,Y)$ should be estimated to determine how many solutions of X dominate Y and vice versa [42].

3.4.5 Hypervolume

The Hypervolume criterion is defined as the volume covered by solutions of set Q in the objective space for a multi-objective problem having two objective functions [60, 61]. This metric evaluates both convergence and diversity of an algorithm. The Hypervolume (HV) metric is determined, according to Eq. (42):

$$\mathrm{HV} = \bigcup_{i=1}^{|Q|} v_i \tag{42}$$

where v_i is a hypercube, which is formed with a reference point for each solution $i \in Q$ and the solution i as the diagonal corners of this hypercube. The reference point is attained by creating a vector of the worst objective function values.

A multi-objective algorithm with a high value for the HV metric is desirable [62].

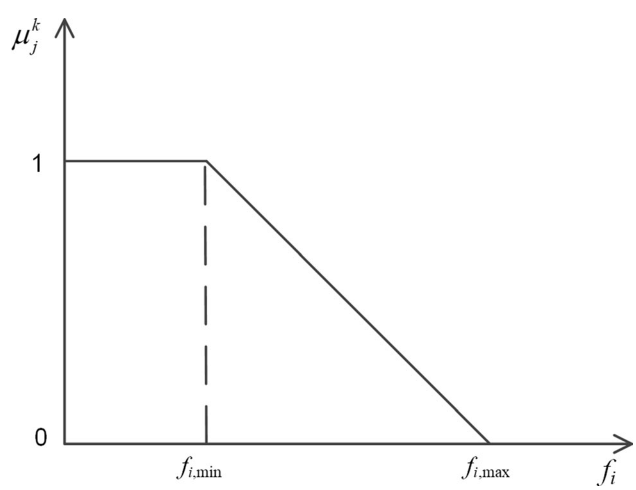


Fig. 5 Fuzzy membership function

3.5 Decision-making method

The essential need for making a decision is to determine the best compromise solution from a non-dominated set [63]. The best compromise solution was defined by applying a fuzzy membership function based on the Fuzzy set theory. First, a linear membership function $\mu_j(k)$ denotes the satisfaction degree of the kth solution for the jth objective function as follows:

$$\mu_j(k) = \begin{cases} 1 & \text{if } f_j(k) \le f_j^{\min} \\ \dfrac{f_j^{\max} - f_j(k)}{f_j^{\max} - f_j^{\min}} & \text{if } f_j^{\min} < f_j(k) < f_j^{\max} \\ 0 & \text{if } f_j(k) \ge f_j^{\max} \end{cases} \tag{43}$$

where $f_j^{\min}$ and $f_j^{\max}$ are minimum and maximum values of the jth objective function in the non-dominated set, respectively. Figure 5 depicts a schematic view of Fuzzy membership function. The value of the fuzzy membership function is within the range [0,1].

Each non-dominated solution has a fuzzy membership for each objective function. The overall membership value of a non-dominated solution is calculated by summing the membership values for all objective functions. The normalized membership function $\mu(k)$ of the kth non-dominated solution is defined as:

Table 2 Mathematical formulations for multi-objective benchmark problems [56]

Problem	Objective functions	Variable range
ZDT1	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1 - \sqrt{x_1/g(x)}\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{D} x_i\right)/(D-1)$	$D = 30$ $0 \le x_i \le 1$ $i = 1, 2, \ldots, D$
ZDT2	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1 - (x_1/g(x))^2\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{D} x_i\right)/(D-1)$	$D = 30$ $0 \le x_i \le 1$ $i = 1, 2, \ldots, D$
ZDT3	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)}\sin(10\pi x_1)\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{D} x_i\right)/(D-1)$	$D = 30$ $0 \le x_i \le 1$ $i = 1, 2, \ldots, D$
ZDT6	$f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$ $f_2(x) = g(x)\left[1 - (f_1(x)/g(x))^2\right]$ $g(x) = 1 + 9\left[\left(\sum_{i=2}^{D} x_i\right)/(D-1)\right]^{0.25}$	$D = 10$ $0 \le x_i \le 1$ $i = 1, 2, \ldots, D$
SCH	$f_1(x) = x^2$ $f_2(x) = (x-2)^2$	$-10^{-3} \le x \le 10^3$
FON	$f_1(x) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i - \frac{1}{\sqrt{3}}\right)^3\right)$ $f_2(x) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i + \frac{1}{\sqrt{3}}\right)^3\right)$	$D = 3$ $-4 \le x_i \le 4$ $i = 1, 2, \ldots, D$
POL	$f_1(x) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$ $f_2(x) = (x_1 + 3)^2 + (x_2 + 1)^2$ $A_1 = 0.5\sin 1 - 2\cos 1 + \sin 2 - 1.5\cos 2$ $A_2 = 1.5\sin 1 - \cos 1 + 2\sin 2 - 0.5\cos 2$ $B_1 = 0.5\sin x_1 - 2\cos x_1 + \sin x_2 - 1.5\cos x_2$ $B_2 = 1.5\sin x_1 - \cos x_1 + 2\sin x_2 - 0.5\cos x_2$	$D = 2$ $-\pi \le x_i \le \pi$ $i = 1, 2, \ldots, D$
KUR	$f_1(x) = \sum_{i=1}^{D-1}\left(-10\exp\left(-0.2\sqrt{x_i^2 + x_{i+1}^2}\right)\right)$ $f_2(x) = \sum_{i=1}^{D}\left(\lvert x_i\rvert^{0.8} + 5\sin x_i^3\right)$	$D = 3$ $-5 \le x_i \le 5$ $i = 1, 2, \ldots, D$

Table 3 Parameters for multi-objective algorithms

MOSGA	NSGA-II	MOMVO	MOPSO
Population size (n_{pop}) = 100	Population size = 100	Population size = 100	Population size = 100
Number of search group members (n_g) = 20	Crossover operator = 20	Inertia weight = 0.4	Worm hole existence probability max = 1
Number of mutations (n_{mut}) = 5	Mutation operator = 20	Adaptive grid = 30	Worm hole existence probability min = 0.2
Perturbation constant (α^k) = 3		Mutation rate = 0.5	

$$\mu(k) = \frac{\sum_{j=1}^{n_{obj}} \mu_j(k)}{\sum_{k=1}^{n_{pf}} \sum_{j=1}^{n_{obj}} \mu_j(k)} \tag{44}$$

where n_{obj} and n_{pf} are numbers of objective functions and non-dominated solutions, respectively.

The solution with a maximal value of the normalized membership function is the best compromise one.

3.6 Implementation of MOSGA for TEO-FPSC problem

To implement the MOSGA to TEO-FPSC problem, each individual of the initial population **P** representing the design variables is defined as follows:

$$P_i = \left[\dot{m}^i, D_o^i, N_t^i, \delta_b^i, \phi^i\right]^T \quad \text{for } i = 1, \ldots, n_{pop} \tag{45}$$

The procedures for TEO-FPSC using MOSGA are stated below.

Step 1: Define the input data, including specifications of FPSC, fluid and material properties, test conditions;
Step 2: Set parameters for the proposed MOSGA;
Step 3: Initial population **P** is randomly generated using Eq. (34);
Step 4: Estimate objective function values for each individual of **P** using Eqs. (1, 20);
Step 5: Sort individuals of population **P** into different non-dominated levels using elitist non-dominated sorting

Table 4 Statistical results of multi-objective algorithms: ZDT1, ZDT2, ZDT3, ZDT6

Algorithm	GD		SP		Δ		Times (s)
	Average	SD	Average	SD	Average	SD	
ZDT1							
MOSGA	**2.1388E−04**	**2.4520E−05**	**6.9071E−03**	**7.2774E−04**	**3.4645E−01**	**2.3452E−02**	0.794
NSGA-II	1.0894E−01	1.0648E−02	2.6192E−02	9.8036E−03	8.4067E−01	2.6313E−02	35.517
MOMVO	1.6091E−02	4.2474E−03	1.2190E−02	3.8905E−03	9.1683E−01	6.6177E−02	**0.786**
MOPSO	1.0011E−01	2.8438E−02	2.9991E−02	1.1661E−02	8.6656E−01	4.6772E−02	1.439
ZDT2							
MOSGA	**2.2760E−04**	**1.6838E−05**	**7.2410E−03**	**5.5655E−04**	**3.6837E−01**	2.7393E−02	0.791
NSGA-II	1.7040E−01	1.4631E−02	1.1070E−02	4.2953E−03	8.8977E−01	**2.4131E−02**	43.431
MOMVO	2.3604E−02	9.4923E−03	2.7001E−02	2.3187E−02	1.0375	5.1571E−02	**0.524**
MOPSO	1.2138E−01	8.1641E−02	8.8471E−03	9.3560E−03	9.4787E−01	7.2642E−02	0.783
ZDT3							
MOSGA	**3.2405E−04**	**2.2082E−05**	**7.9654E−03**	**8.4973E−04**	**7.0070E−01**	**1.5167E−02**	1.023
NSGA-II	1.0837E−01	1.8851E−02	2.0680E−02	1.6381E−02	9.0124E−01	5.8144E−02	35.775
MOMVO	1.7415E−02	8.0868E−03	2.3266E−02	5.0130E−02	9.7116E−01	6.8850E−02	**0.747**
MOPSO	1.1159E−01	2.2696E−02	3.2828E−02	1.7456E−02	8.7848E−01	3.1208E−02	1.482
ZDT6							
MOSGA	**4.1767E−04**	**2.3922E−05**	**6.8021E−03**	**5.0801E−04**	**4.0252E−01**	**2.6440E−02**	0.713
NSGA-II	2.7738E−01	6.0877E−02	4.3807E−02	3.7516E−02	1.0653	7.2934E−02	42.094
MOMVO	5.3318E−02	3.9977E−02	2.1269E−01	1.0979E−01	1.1176	1.4065E−01	**0.293**
MOPSO	1.2526E−01	6.3961E−02	2.3512E−01	2.0441E−01	1.2317	5.7609E−02	0.821

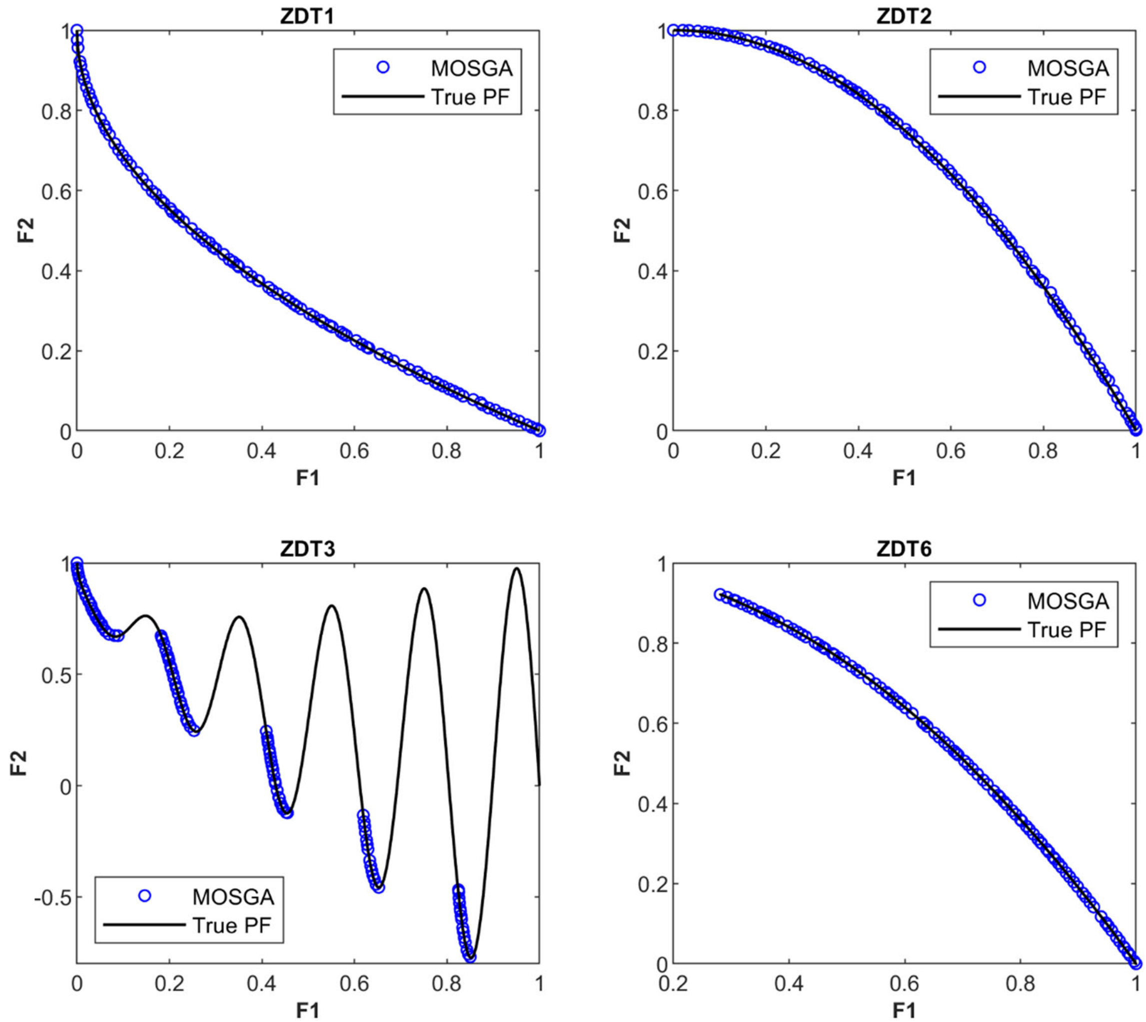


Fig. 6 Pareto optimal fronts generated by MOSGA: ZDT1, ZDT2, ZDT3, ZDT6

technique and store all individuals of **P** in a Pareto archive;

Step 6: Generate initial search group $\mathbf{R}^k$ choosing n_g solutions from **P** employing tournament selection;

Step 7: Mutate n_{mut} individuals by newly generated members by Eq. (35);

Step 8: Generate families ($\mathbf{F}_i$) according to Eq. (36) and save newly created solutions in the advanced Pareto archive;

Step 9: Combine the current archive and the advanced archive;

Step 10: Select best solutions for entry into the new Pareto archive based on the Pareto archive selection mechanism;

Step 11: Select a new search group as follows:

- Global phase: search group $\mathbf{R}^{k+1}$ is created by the best member of each family;
- Local phase: search group $\mathbf{R}^{k+1}$ is created by the best n_g solutions from Pareto archive.

Step 12: Update α^{k+1} using Eq. (37);

Step 13: Set $k = k + 1$, if $k > it^{\max}$—move to Step 14; if otherwise back to Step 7.

Step 14: Solutions obtained: Pareto optimal solutions in the final Pareto archive.

Step 15: Extract best compromise solutions using the decision-making method as given in Sect. 3.5.

4 Simulation results

4.1 Multi-objective benchmark test problems

Eight well-known benchmark problems with diverse features were used to evaluate the capability and performance of the proposed MOSGA. These problems were selected from credible research studies, including Zitzler–Deb–Thiele's functions (ZDT1, ZDT2, ZDT3, and ZDT6) [64], Schaffer's function (SCH) [65], Fonseca and Fleming's

Table 5 Statistical results of multi-objective algorithms: SCH, FON, POL, KUR

Algorithm	GD		SP		Δ		Times (s)
	Average	SD	Average	SD	Average	SD	
SCH							
MOSGA	**1.8831E−04**	**7.9801E−06**	**2.5942E−02**	**1.6471E−03**	**3.7387E−01**	2.5347E−02	0.802
NSGA-II	2.7508E−04	6.4662E−05	2.9235E−02	2.7557E−03	4.3075E−01	**1.8947E−02**	42.050
MOMVO	2.9784E−04	1.8231E−04	3.9977E−02	5.4336E−03	7.1467E−01	4.0774E−02	**0.363**
MOPSO	4.3797E−04	1.8952E−04	2.7334E−02	3.4520E−03	4.0351E−01	3.4225E−02	4.715
FON							
MOSGA	**1.8032E−04**	**1.5208E−05**	**6.2334E−03**	6.0944E−04	3.5381E−01	**1.2810E−02**	**0.930**
NSGA-II	2.2304E−04	2.8432E−05	6.4277E−03	**3.9243E−04**	**3.4697E−01**	1.9473E−02	37.482
MOMVO	3.1679E−04	1.4978E−04	1.4458E−02	1.9185E−03	1.0639	5.8118E−02	1.092
MOPSO	4.3903E−04	8.2173E−05	7.3628E−03	1.6136E−03	4.2587E−01	4.8291E−02	4.062
POL							
MOSGA	**1.5265E−03**	**1.9804E−04**	**8.4134E−02**	**3.8502E−03**	**9.5188E−01**	**4.9722E−03**	**0.709**
NSGA-II	8.2963E−03	1.5859E−02	1.0568E−01	5.8525E−02	9.7817E−01	1.6996E−02	37.917
MOMVO	8.3337E−03	1.6133E−02	1.8453E−01	4.2654E−02	1.4366	3.7651E−02	1.218
MOPSO	2.2274E−02	1.7102E−02	2.0395E−01	1.1083E−01	1.0210	3.5537E−02	2.764
KUR							
MOSGA	**2.0073E−03**	**1.4231E−04**	**5.7580E−02**	**3.7422E−03**	**4.6631E−01**	1.3403E−02	0.844
NSGA−II	2.1704E−03	2.4952E−04	8.7629E−02	1.6940E−02	4.7691E−01	**1.1861E−02**	33.683
MOMVO	4.1695E−03	1.2811E−03	1.3174E−01	3.2966E−02	9.5054E−01	6.5042E−02	**0.805**
MOPSO	1.4806E−02	5.9938E−03	1.5643E−01	5.7412E−02	7.5696E−01	1.1152E−01	1.956

The best results are highlighted in bold

function (FON) [66], Poloni's function (POL) [67], and Kursawe's function (KUR) [68].

All mathematical formulations of these test functions are stated in Table 2. The MOSGA was performed in MATLAB programming software. Outcomes were compared with those from the following multi-objective algorithms: (1) non-dominated sorting genetic algorithm II (NSGA-II) [54]; (2) multi-objective multi-verse optimizer (MOMVO) [43]; and (3) multi-objective particle swarm optimization (MOPSO) [69]. The number of function evaluations (NFEs) was 10,000 (stop criteria) for reliable comparisons. All algorithms were independently run 30 times (N_{trials} = 30) for each test case to analyze and assure robust statistical performance. Table 3 summarizes the parameters for four algorithms. Final results were compared with each other based on three performance metrics (GD, SP, and Δ).

4.1.1 Analysis of results

Tables 4 summarizes the statistical results, including average and standard deviation (SD) values of three performance metrics (GD, SP, and Δ) and computation times for all algorithms for ZDT test suites. The best results are highlighted in bold. Table 4 shows that MOSGA outperformed the other techniques for average GD values and the stability of generated solutions (lower SD). On the other hand, NSGA-II, MOMVO, and MOPSO all failed to seek Pareto optimal solutions near true Pareto optimal front. This was confirmed by their high GD values. Moreover, MOSGA not only obtained minimum GD metric but also placed the first rank for SP and Δ metrics. From the evaluation, it can be concluded that the proposed MOSGA yielded the best results for ZDT test suites.

Figure 6 illustrates Pareto optimal fronts generated by the proposed MOSGA for ZDT test suites. These figures clearly showed that MOSGA successfully converged on true Pareto front with proper distribution and spread of solutions.

Table 5 presents the statistical results of three performance metrics (GD, SP, and Δ) for test functions SCH, FON, POL, and KUR. According to Table 5, it can be inferred that the MOSGA had the best statistical results for GD metric. Similarly, MOSGA also surpassed the other algorithms for SP and Δ outcomes. Regarding all metrics in Table 5, NSGA-II proved second-best by generating a near-optimal Pareto front with appropriate distributions. However, the computation costs of NSGA-II were the

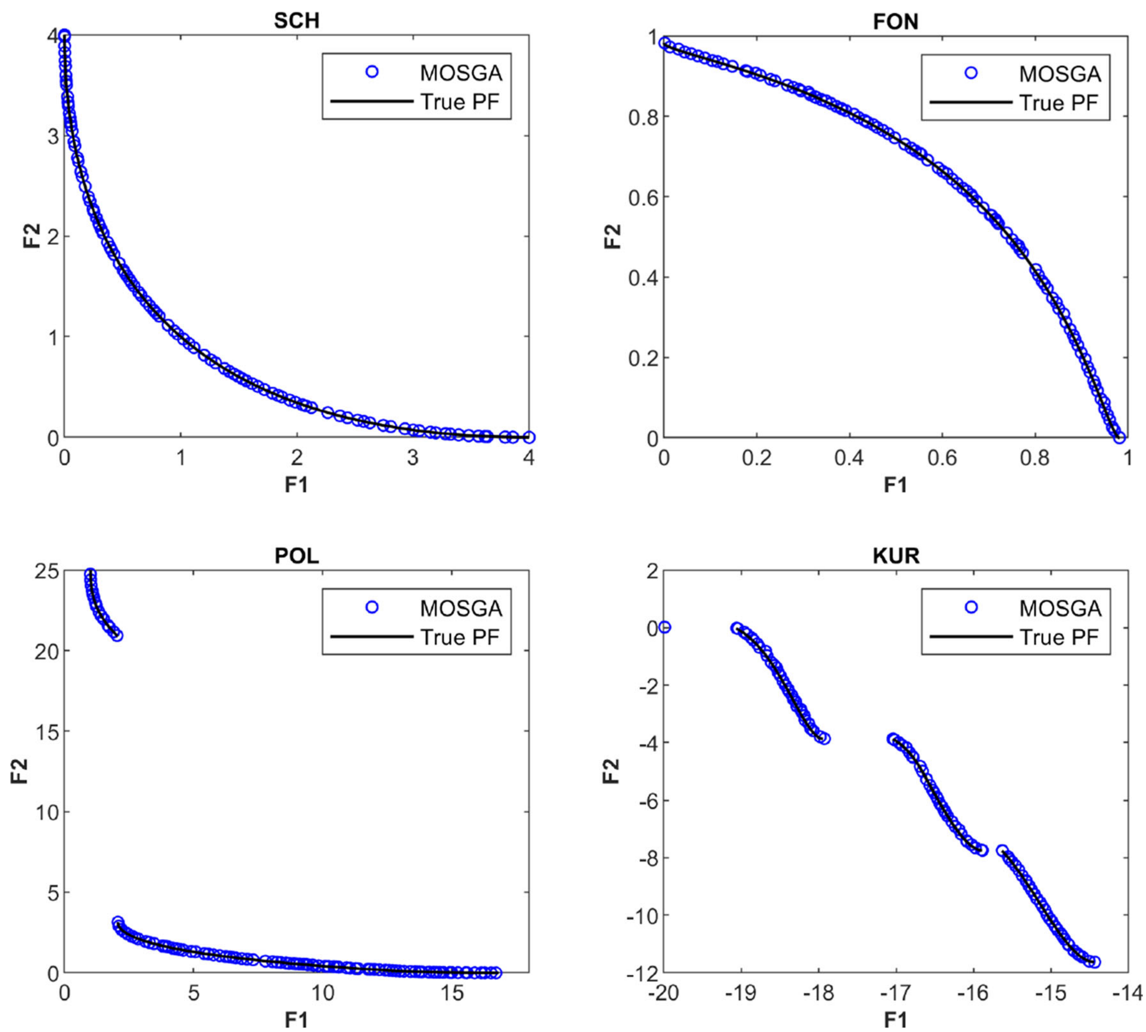


Fig. 7 Pareto optimal fronts generated by MOSGA: SCH, FON, POL, KUR

highest. MOMVO with the highest average values for SP and Δ metrics was unable to produce a well-distributed Pareto front. On the other hand, MOPSO produced the worst statistical results for GD metric. In summary, MOSGA proved superior for these test functions, especially in terms of GD metric.

Figure 7 illustrates Pareto optimal fronts generated by the MOSGA for test functions SCH, FON, POL, and KUR. These figures demonstrated that Pareto optimal fronts obtained by MOSGA not only converged quite well on the true Pareto front but also distributed appropriately. Overall, Pareto optimal solutions found by MOSGA had the highest convergence and best distribution for all test problems.

The average computational times of four methods (MOSGA, NSGA-II, MOMVO, and MOPSO) for each test function are reported in the last columns in Tables 4 and 5. MOSGA had the best computational time for FON and POL test functions. Furthermore, MOSGA's computational times were the second-best among four algorithms for ZDT1, ZDT3, ZDT6, SCH, and KUR test functions. Although MOSGA's computational times were not the best, MOSGA provided better quality solutions than NSGA-II, MOMVO, and MOPSO for most of the test functions.

4.1.2 Robustness analysis

In order to further analyze the performance of the proposed MOSGA, Figs. 8 and 9 depict the box plots of GD, SP, and Δ metrics yielded by each algorithm on each test function for 30 independent runs. All distributions of performance metrics were represented as rectangle boxplots. A red line denoted the mean value. Boundary values, except for outliers, were shown by top and bottom whiskers for each box. Outliers were plotted individually using the (+) symbol. In most test problems, the proposed MOSGA had the box plots with a smaller rectangle and a lower red line than NSGA-II, MOMVO, and MOPSO methods. This indicated

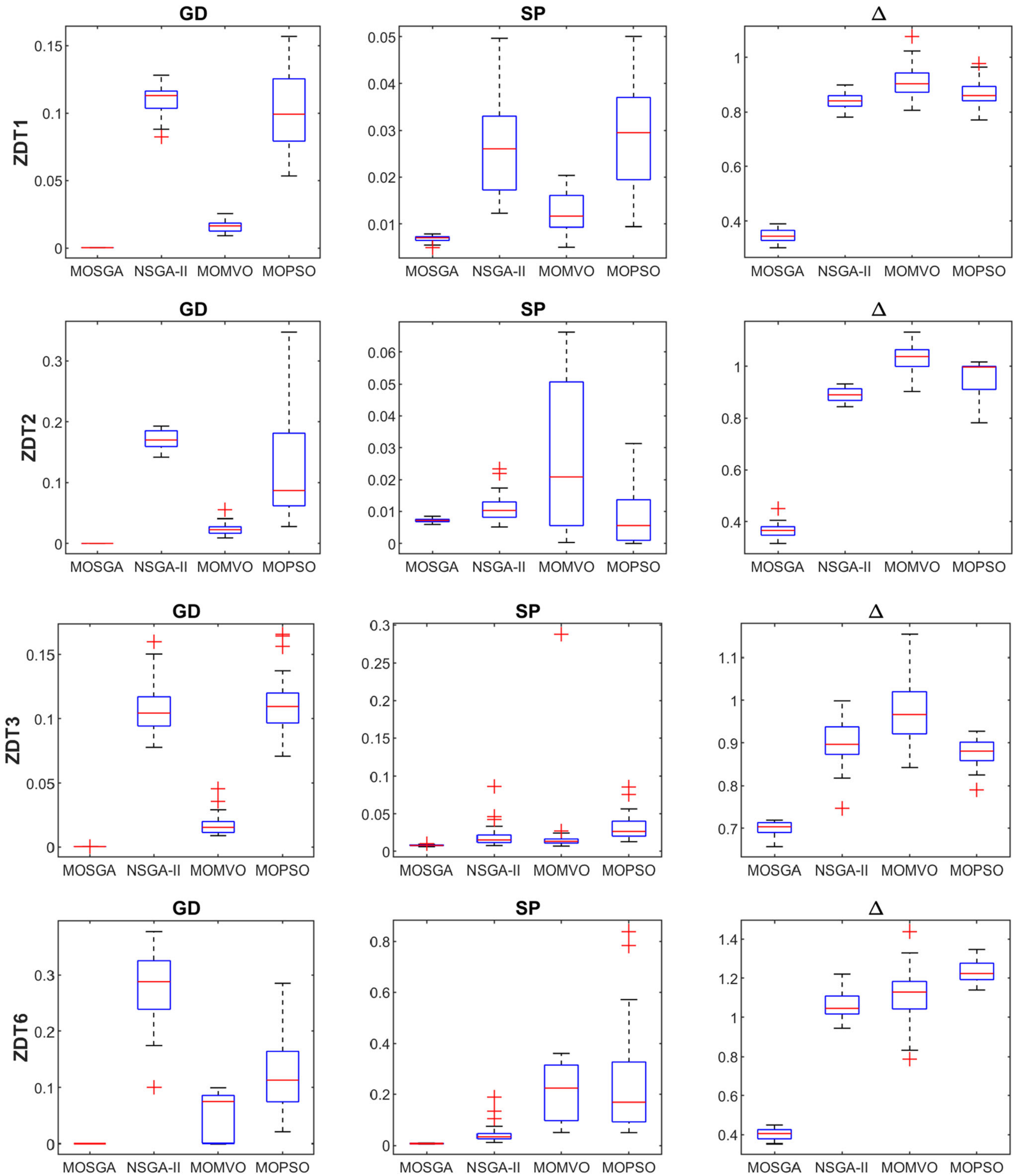


Fig. 8 Box plots of performance metrics (GD, S, and Δ): ZDT1, ZDT2, ZDT3, ZDT6

that the stability and robustness of MOSGA had obvious advantages over other compared algorithms.

4.1.3 Statistical test

To verify whether the results of MOSGA are significantly superior to the results of other algorithms or not, a

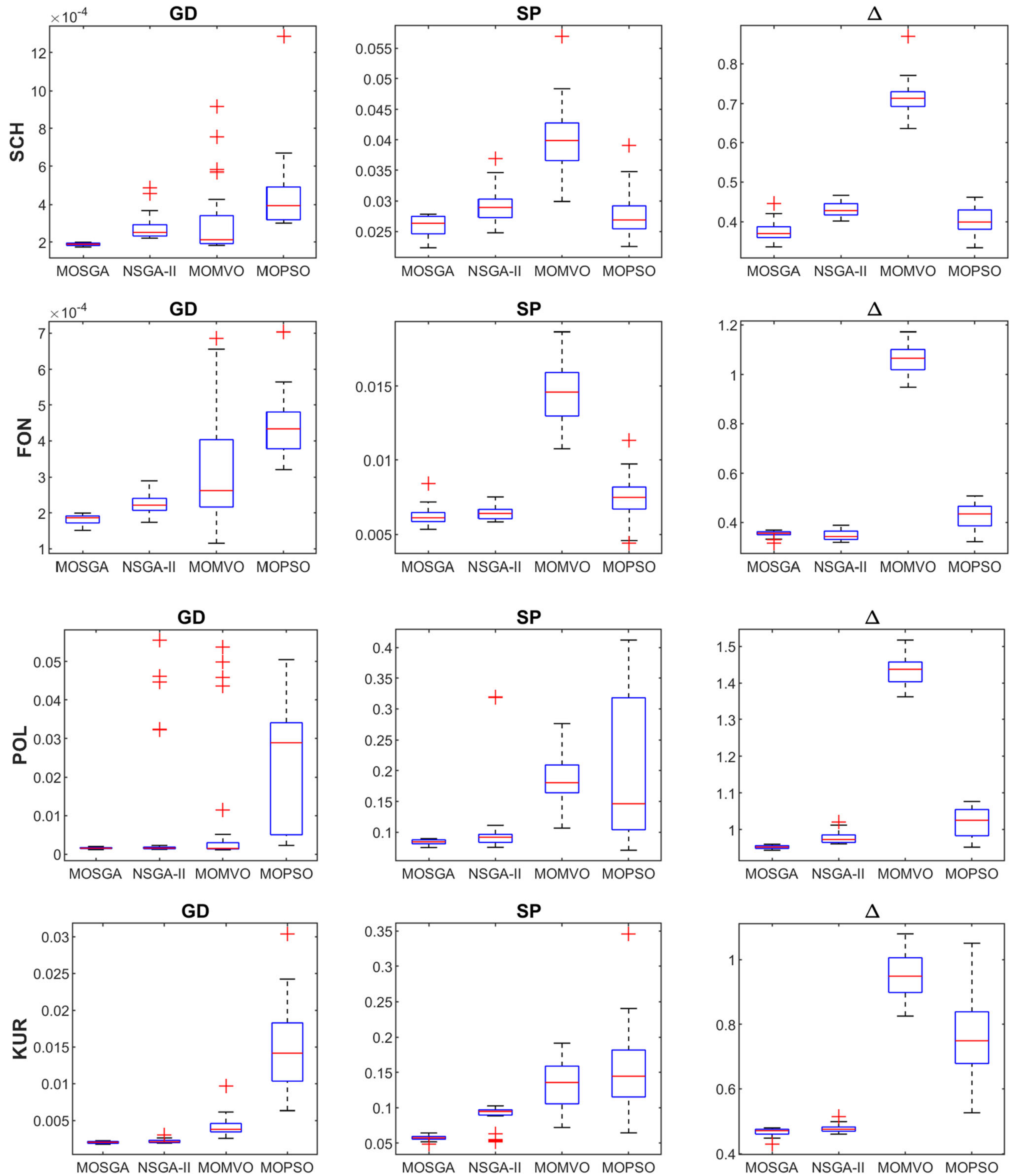


Fig. 9 Box plots of performance metrics (GD, S, and Δ): SCH, FON, POL, KUR

nonparametric statistical test, called the Wilcoxon rank-sum test, was conducted for 30 independent runs at a significance level of 5%. Tables 6, 7, and 8 show the Wilcoxon rank-sum test results for all test functions in terms of GD, SP, and Δ metrics. A p value < 0.05 and signed with "+" indicates a significant difference between two solution sets of the MOSGA and other algorithms.

Table 6 Wilcoxon rank-sum test results based on the GD metric for all test functions

MOSGA versus	NSGA-II		MOMVO		MOPSO	
	p value	Signed	p value	Signed	p value	Signed
ZDT1	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT2	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT3	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT6	3.02E−11	+	8.35E−08	+	3.02E−11	+
SCH	9.92E−11	+	1.02E−05	+	3.02E−11	+
FON	1.07E−07	+	4.80E−07	+	3.02E−11	+
POL	5.39E−01	−	6.84E−01	−	3.02E−11	+
KUR	1.38E−02	+	3.02E−11	+	3.02E−11	+

Table 7 Wilcoxon rank-sum test results based on the SP metric for all test functions

MOSGA versus	NSGA-II		MOMVO		MOPSO	
	p value	Signed	p value	Signed	p value	Signed
ZDT1	3.02E−11	+	2.92E−09	+	3.02E−11	+
ZDT2	3.57E−06	+	2.38E−03	+	2.64E−01	−
ZDT3	6.72E−10	+	1.01E−08	+	3.02E−11	+
ZDT6	3.02E−11	+	3.02E−11	+	3.02E−11	+
SCH	3.57E−06	+	3.02E−11	+	1.45E−01	−
FON	3.27E−02	+	3.02E−11	+	4.71E−04	+
POL	4.71E−04	+	3.02E−11	+	7.12E−09	+
KUR	2.68E−06	+	3.02E−11	+	3.02E−11	+

Table 8 Wilcoxon rank-sum test results based on the Δ metric for all test functions

MOSGA versus	NSGA-II		MOMVO		MOPSO	
	p value	Signed	p value	Signed	p value	Signed
ZDT1	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT2	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT3	3.02E−11	+	3.02E−11	+	3.02E−11	+
ZDT6	3.02E−11	+	3.02E−11	+	3.02E−11	+
SCH	2.03E−09	+	3.02E−11	+	4.46E−04	+
FON	1.45E−01	−	3.02E−11	+	2.39E−08	+
POL	3.02E−11	+	3.02E−11	+	1.78E−10	+
KUR	1.03E−02	+	3.02E−11	+	3.02E−11	+

As per the Wilcoxon rank-sum test results for GD metrics in Table 6, MOSGA obtained statistically different results from NSGA-II, MOMVO, and MOPSO in seven, seven, and eight test functions, respectively, out of eight test functions. From Table 7, results of the MOSGA were statistically different from the results of the NSGA-II, MOMVO, and MOPSO in eight, eight, and six test functions, respectively, in terms of Wilcoxon rank-sum test results for SP metric. Likewise, Table 8 shows statistical test results for Δ metric, MOSGA was different from NSGA-II, MOMVO, and MOPSO in seven, eight, and eight test functions, respectively, out of eight test functions. It was apparent from the statistical results that the proposed MOSGA had significantly better performance than other algorithms in most of the test problems.

4.2 TEO-FPSC problem

In this research work, the specifications of FPSC were referred to the technical details of Kingspan solar collector FPW25 [70]. An aluminum sheet with an emissivity of 0.05 and thermal conductivity of 240 W/m K was used as the absorber plate. According to available market prices for components, constants for a_i, $b_{i,}$ and all collector assembly coefficients were [120 60 220 4.5 3500], [0.9 0.8 1 1 0.47], and 1.5, respectively, as proposed by Hajabdollahi et al.

Table 9 Specifications and test conditions of FPSC

Parameter	Value
Cover surface area (A_c)	2.4213 m^2
Absorber plate area (A_p)	2.2388 m^2
Absorber plate thickness (δ)	0.3 mm
Emissivity of glass cover (ε_g)	0.84
Emissivity of absorber plate (ε_p)	0.04
Effective transmittance-absorptance ($\tau\alpha$)	0.8645
Length of tube (L_t)	1.94 m
Fluid inlet temperature (T_i)	10 °C
Ambient temperature (T_a)	10 °C
Total solar radiation intensity (I_T)	1000 W/m^2
Slope of collector (β)	20°
Wind speed (v)	5 m/s

[29]. The unit price for electricity was estimated at 0.10 $/kWh for a system that operated approximately 4,380 h annually. The lifetime of FPSC was estimated at 15 years, with an inflation rate of 12%. Table 9 summarizes all characteristics and test conditions of FPSC.

For the same input values given in Table 9, simulation results were verified by comparisons with the corresponding results published by the manufacturer [70] as presented in Table 10. Differences in percentage values for both results' sets under different inlet temperatures were acceptable.

4.2.1 Optimization results

The proposed MOSGA was implemented for the TEO-FPSC problem. To generate a broad spectrum of optimal results, optimization procedures were conducted for working fluids of pure water and SiO_2, Al_2O_3, CuO nanofluids. The initial parameters of MOSGA (n_{pop}, n_g, n_{mut}, and α^k) were selected as 100, 20, 5, and 3, respectively. Other control parameters were set as follows: the NFEs was 10,000; the number of Pareto optimal solutions was 100, and the number of trials was 30.

Pareto optimal fronts obtained by MOSGA for all working fluids are shown in Fig. 10(a–d). They described the relationship between efficiency and TAC and were depicted as trade-off curves that trace actual conflicts between both objective functions. Increasing efficiency followed increasing TAC, and the slope steeply increased when approaching the highest feasible efficiency.

Table 11 shows the optimal values of efficiency and TAC for Solutions A–B–C. Solution A yielded maximum values for efficiency and TAC, defined as the best thermodynamic optimized point. Solution C yielded minimum values for both objectives, defined as the best economic optimized point. Moreover, the decision-making method was used to determine the best compromise solution (Solution B) from the Pareto optimal set. Thus, Solution B struck a balance between both objectives, defined as the best thermo-economic optimized point. In fact, solution A was the optimal solution in single-objective optimization for efficiency, whereas solution C was the optimal solution in single-objective optimization for the TAC.

Table 12 lists values of objective functions and design variables for the best thermo-economic optimized points (as depicted in Fig. 10). These results indicated that improved efficiency rates were 2.2748%, 2.4298%, and 2.7948% for SiO_2, Al_2O_3, and CuO, respectively, compared to pure water. Meanwhile, TAC rates were increased by 2.4111%, 2.3403%, and 2.9133%, respectively.

Based on obtained solutions in Pareto optimal fronts, decision-makers can finalize solution for a particular project on the basis of experience and desired goals under specific circumstances. If the priority is the performance of FPSC, solution A will be the best solution. If the priority is budget, solution C will be optimal. Additionally, if a manufacturer prefers a measurable balance among objectives, solution C will offer the best compromise providing acceptable efficiency and an affordable cost for FPSC systems.

To illustrate the concept of domination, Fig. 11a shows Pareto optimal fronts generated for all four studies. As a key observation, all Pareto fronts of all nanofluids dominated pure water. For further analysis, Table 13 shows comparisons with respect to the C-metric in which $F1$, $F2$, $F3$, and $F4$ denote Pareto fronts for pure water, SiO_2, Al_2O_3, and CuO, respectively. Specifically, 95.03%, 96.28%, and 95.30% solutions of pure water were dominated by SiO_2, Al_2O_3, and CuO, respectively, on average. Thus, nanofluids proved better than pure water. Table 13 also indicates that CuO nanoparticles were superior by

Table 10 Comparison of simulation results with manufacturer's reference [70]

Output parameters	$T_i - T_a$ (°C)				
	0	10	30	50	70
Power output of manufacturer [70] (W)	1741	1659	1480	1283	1068
Power output of this work (W)	1712.09	1625.89	1453.52	1249.84	1025.97
Difference (%)	− 1.66	− 1.99	− 1.79	− 2.58	− 4.02

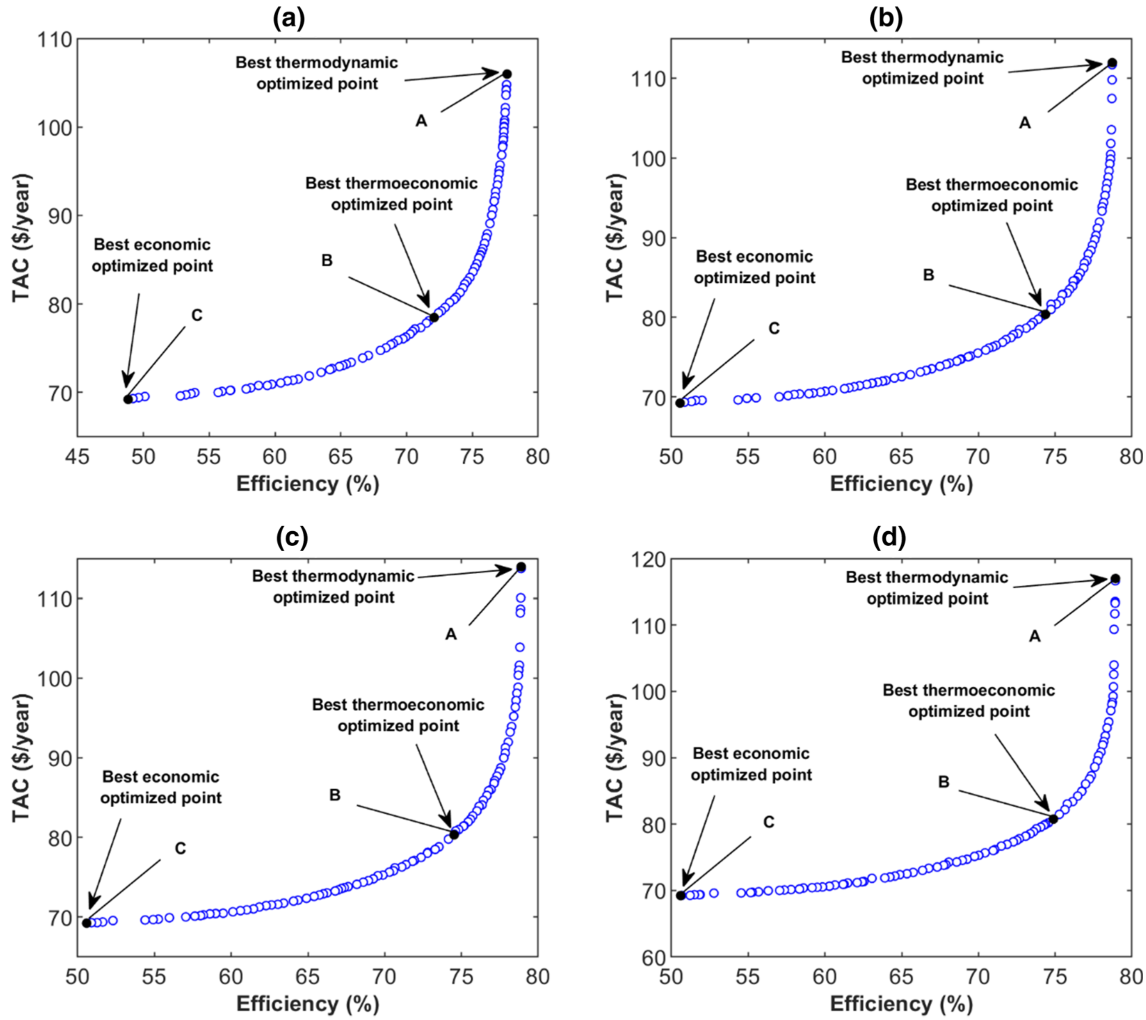


Fig. 10 Pareto optimal fronts for working fluids: **a** pure water; **b** SiO_2; **c** Al_2O_3; **d** CuO

Table 11 Optimal values of efficiency and the TAC for solution A–B–C in Pareto optimal front in Fig. 10a–d

Case study	Best thermodynamic optimized point (Solution A)		Best thermo-economic optimized point (Solution B)		Best economic optimized point (Solution C)	
	Efficiency (%)	TAC ($/year)	Efficiency (%)	TAC ($/year)	Efficiency (%)	TAC ($/year)
Pure water	77.6388	105.9559	72.0910	78.4756	48.8338	69.2102
SiO_2	78.7209	111.9504	74.2373	80.1317	50.5639	69.2166
Al_2O_3	78.8924	113.9641	74.5208	80.3122	50.5838	69.2171
CuO	78.9288	116.9708	74.8858	80.7618	50.5957	69.2174

more than 46.24% over Al_2O_3 (ranked second) and 78.93% over SiO_2 (ranked third) results, on average.

Tables 14 and 15 depict values of objective functions and design parameters for two design points, D and E, in Fig. 11b. For a fixed TAC of 100 $/year (design point D in Fig. 11b), efficiencies for SiO_2, Al_2O_3, and CuO increased 1.2072%, 1.2972%, and 1.3879%, respectively, compared to pure water. For a fixed efficiency at 75% (design point E in Fig. 11b), TAC for SiO_2, Al_2O_3, and CuO decreased 2.3842%, 2.7034%, and 2.6113%, respectively, compared to pure water.

The efficiency improvement of FPSC systems as using nanofluids is expected. Nanofluids intensify thermal conductivity, diffusivity, and the convection heat transfer

Table 12 Values of objective functions and design parameters for the best thermo-economic optimized point

Parameter	Pure water	SiO_2	Al_2O_3	CuO
Thermal efficiency (%)	72.0910	74.3658	74.5208	74.8858
Total annual cost ($/year)	78.4756	80.3677	80.3122	80.7618
Mass flow rate (kg/s)	0.0223	0.0254	0.0294	0.0277
Tube diameter (mm)	0.0050	0.0050	0.0050	0.0050
Tube number	17	19	17	18
Insulator thickness (mm)	0.0397	0.0420	0.0322	0.0378
Particle volumetric concentration	–	0.0223	0.0476	0.0661

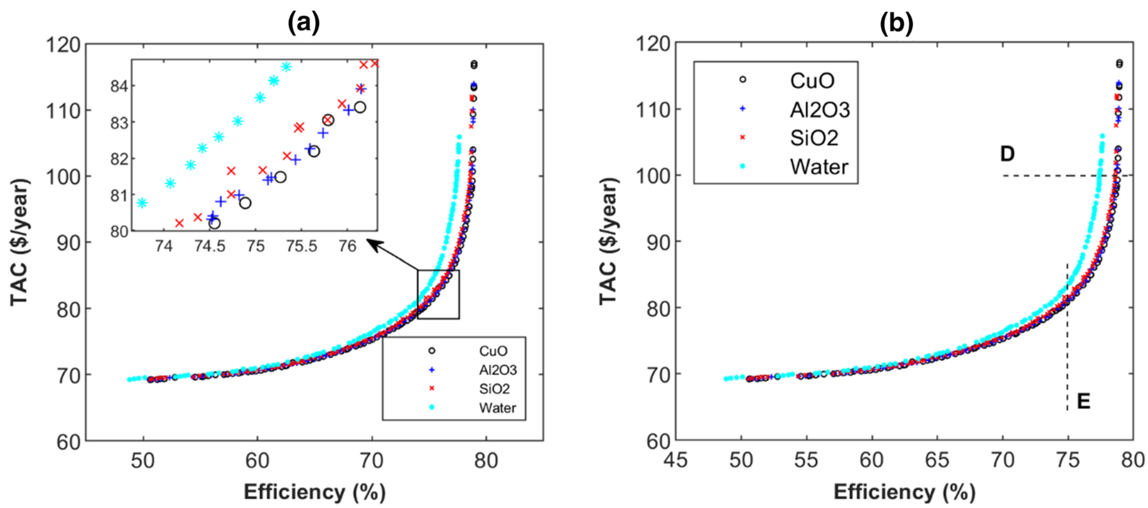


Fig. 11 Concept of domination for Pareto optimal fronts of four case studies

Table 13 Comparison of *C*-metric for four case studies of different working fluids

Performance measurement	$C(F2, F1)$	$C(F3, F1)$	$C(F3, F2)$	$C(F4, F1)$	$C(F4, F2)$	$C(F4, F3)$
Best	0.9900	0.9900	0.9000	0.9900	0.9600	0.6900
Average	0.9503	0.9628	0.7047	0.9530	0.7893	0.4624
Worst	0.8800	0.9000	0.5000	0.8000	0.5200	0.2500
SD	0.0185	0.0152	0.0724	0.0316	0.0792	0.0657

Table 14 Values of objective functions and design parameters for design point D (Fig. 11b) with TAC $= 100 \pm 0.7\%$ $/year

Parameter	Pure water	SiO_2	Al_2O_3	CuO
Thermal efficiency (%)	77.4059	78.6131	78.7031	78.7938
Total annual cost ($/year)	100.0647	100.4596	100.3605	100.6868
Mass flow rate (kg/s)	0.0982	0.0988	0.1000	0.1000
Tube diameter (mm)	0.0097	0.0080	0.0050	0.0073
Tube number	20	20	20	20
Insulator thickness (mm)	0.0989	0.0945	0.1000	0.0989
Particle volumetric concentration	–	0.1000	0.0845	0.1000

coefficient. Consequently, heat transfer rates increase and result in enhanced efficiency. On the other hand, any decrease in TAC as using nanofluids is unpredictable. A higher TAC is expected to be compared to pure water due to the high price of nanoparticles. Nonetheless, adding nanoparticles to pure water reduces radiation and convection losses due to increased heat transfer between the absorption plate and the nanofluid. Therefore, a lower mass flow rate related to pump usage is needed to reduce operational costs while achieving the desired performance. As a result, different nanofluids simultaneously enhance efficiency and desirable TAC results.

Table 15 Values of objective functions and design parameters for design point E (Fig. 11b) with $\eta = 75\% \pm 0.3\%$

Parameter	Pure water	SiO_2	Al_2O_3	CuO
Thermal efficiency (%)	75.0489	75.0751	75.1298	75.2698
Total annual cost ($/year)	83.6632	81.6685	81.4014	81.4785
Mass flow rate (kg/s)	0.0387	0.0283	0.0277	0.0293
Tube diameter (mm)	0.0050	0.0051	0.0050	0.0050
Tube number	20	20	20	19
Insulator thickness (mm)	0.0428	0.0449	0.0389	0.0364
Particle volumetric concentration	–	0.0188	0.0458	0.0642

Table 16 Comparison of four multi-objective algorithms based on *C*-metric

Test case	Indicators	C(*A*1,*A*2)	C(*A*2,*A*1)	C(*A*1,*A*3)	C(*A*3,*A*1)	C(*A*1,*A*4)	C(*A*4,*A*1)
Pure water	Average	0.1791	0.1331	0.2752	0.0741	0.8083	0.0173
	SD	0.0442	0.0453	0.1190	0.0283	0.0929	0.0175
SiO_2	Average	0.1985	0.1613	0.2264	0.0898	0.7641	0.0233
	SD	0.0684	0.0514	0.1002	0.0324	0.1061	0.0211
Al_2O_3	Average	0.2112	0.1626	0.2724	0.0884	0.7162	0.0233
	SD	0.0709	0.0493	0.1187	0.0334	0.1176	0.0218
CuO	Average	0.2155	0.1550	0.2801	0.0934	0.7493	0.0194
	SD	0.0967	0.0609	0.1133	0.0429	0.1073	0.0189

*A*1, *A*2, *A*3 and *A*4 designate MOSGA, NSGA-II, MOMVO, and MOPSO, respectively

Table 17 Comparative results of SP metric among four multi-objective algorithms

Test case	MOSGA	NSGA-II	MOMVO	MOPSO
Pure water				
Average	**0.2575**	0.2967	0.4789	0.6419
SD	**0.0260**	0.0460	0.1058	0.2284
SiO_2				
Average	**0.2922**	0.3836	0.4808	0.6582
SD	**0.0485**	0.0706	0.0849	0.1978
Al_2O_3				
Average	**0.3176**	0.4253	0.5242	0.7591
SD	**0.0541**	0.0713	0.1635	0.2549
CuO				
Average	**0.3530**	0.4786	0.5436	0.7816
SD	**0.0733**	0.1121	0.1606	0.3385

The best results are highlighted in bold

In view of the obtained results, MOSGA was successfully implemented for the TEO-FPSC problem. The level of conflict between the two objectives was revealed as the Pareto optimal front. Set of Pareto optimal solutions provides decision-makers with multiple options for choosing the final solution for a specific project scenario. Moreover, the Pareto optimal fronts for the case with nanoparticles dominated over the case with pure water, in which CuO nanoparticle was the best nanoparticle among the studied nanoparticles.

4.2.2 Statistical comparison and analysis

To evaluate the applicability of the proposed algorithm for the TEO-FPSC problem, MOSGA was run 30 trials independently for each case. Optimization results were compared with other techniques, including NSGA-II, MOMVO, and MOPSO. To assure fair comparisons, the study employed the NFEs of 10,000 and the number of Pareto optimal solutions of 100 for all trials and all four algorithms. Control parameters for all multi-objective algorithms were kept the same values as given in Table 3. All algorithms were compared based on three performance metrics: C-metric, SP, and HV metrics. Results are now reported.

a. C-metric

Table 16 shows C-metric results for all four algorithms. Table 16 demonstrates that MOSGA dominated more than 17.91% of NSGA-II; 27.52% of MOMVO; and 80.83% of MOPSO solutions on average for pure water. For SiO_2, MOSGA dominated 19.85% of NSGA-II; 22.64% of MOMVO; and 74.64% of MOPSO solutions. For Al_2O_3, MOSGA dominated 21.12%, 27.24%, and 71.62% of solutions by NSGA-II, MOMVO, and MOPSO, respectively. For CuO, MOSGA dominated more than 21.55% of NSGA-II;

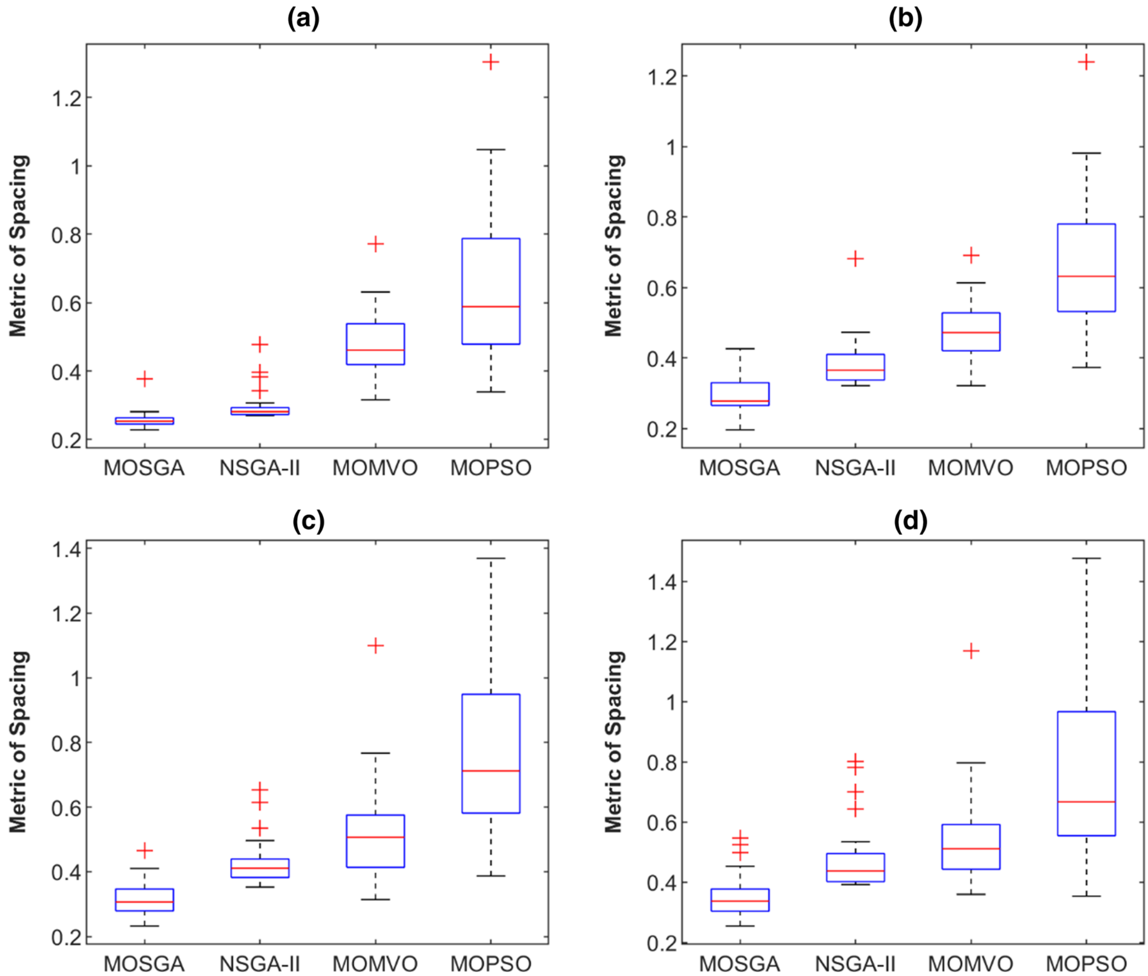


Fig. 12 Box plots of SP metric for four case studies: **a** pure water, **b** SiO_2, **c** Al_2O_3, **d** CuO

Table 18 Wilcoxon rank-sum test results based on the SP metric for the TEO-FPSC problem

MOSGA versus	NSGA-II		MOMVO		MOPSO	
	p value	Signed	*p* value	Signed	*p* value	Signed
Pure water	4.57E−09	+	5.49E−11	+	3.34E−11	+
SiO_2	2.57E−07	+	2.37E−10	+	5.49E−11	+
Al_2O_3	4.69E−05	+	4.57E−09	+	8.99E−11	+
CuO	4.80E−07	+	2.39E−08	+	2.03E−09	+

28.01% of MOMVO; and 74.93% of MOPSO solutions. Therefore, Pareto optimal solutions of MOSGA were superior to the one obtained by MOPSO by far and slightly better than those of NSGA-II and MOMVO.

b. Spacing metric

Table 17 gives a comparison of the SP metric for four multi-objective techniques. Boxplot analyses are depicted in Fig. 12. In all cases, MOSGA yielded the narrowest boxplots placed at the lower extremes of each figure, indicating the range between best and the worst SP values was relatively small as well as least. Furthermore, red lines for all MOSGA boxplots placed lower, indicating minimal medians. This evidence demonstrated a robust performance of the MOSGA in terms of SP metric.

Moreover, the results of the Wilcoxon rank-sum test for the SP metric are provided in Table 18. Based on the results, MOSGA performed significantly better than NSGA-II, MOMVO, and MOPSO for all cases. Therefore, MOSGA could obtain Pareto optimal solutions with the best distribution.

c. Hypervolume metric

Table 19 summarizes results in terms of the HV

Table 19 Comparison of HV metric among all four multi-objective algorithms

Test case	MOSGA	NSGA-II	MOMVO	MOPSO
Pure water				
Average	**1497.3962**	1496.3508	1483.8042	1453.2877
SD	**0.5872**	1.4242	2.6730	7.0889
SiO_2				
Average	**1551.5233**	1548.7586	1538.9742	1504.2047
SD	**0.7443**	1.9243	2.6219	10.4321
Al_2O_3				
Average	**1560.0030**	1555.9182	1546.7074	1511.2856
SD	**0.7288**	3.1913	4.0629	10.2504
CuO				
Average	**1562.4491**	1554.3285	1550.6383	1513.8888
SD	**1.7019**	2.7927	1.9647	10.8484

The best results are highlighted in bold

metric. For comparisons, the same reference point *W* was employed in all trials. Figure 13 illustrates boxplot analyses. MOSGA yielded the narrowest boxplots in the uppermost extremes of each figure. Moreover, red lines for the MOSGA boxplots were also higher, indicating a robust performance with the highest mean values. Table 19 and Fig. 13 show that MOSGA produced the highest HV values for all cases.

Table 20 presents the results of the Wilcoxon rank-sum test for HV metric. Table 20 shows that the MOSGA had statistically better performance than other algorithms for HV metric for all cases. Therefore, it could be concluded that MOSGA proved superior convergence and diversity of Pareto optimal solutions in comparison with NSGA-II, MOMVO, and MOPSO.

From the assessments of C-metric, SP, and HV metrics, it could be concluded that the MOSGA was effectively applied to solving the TEO-FPSC problem with high solution quality. For all case studies,

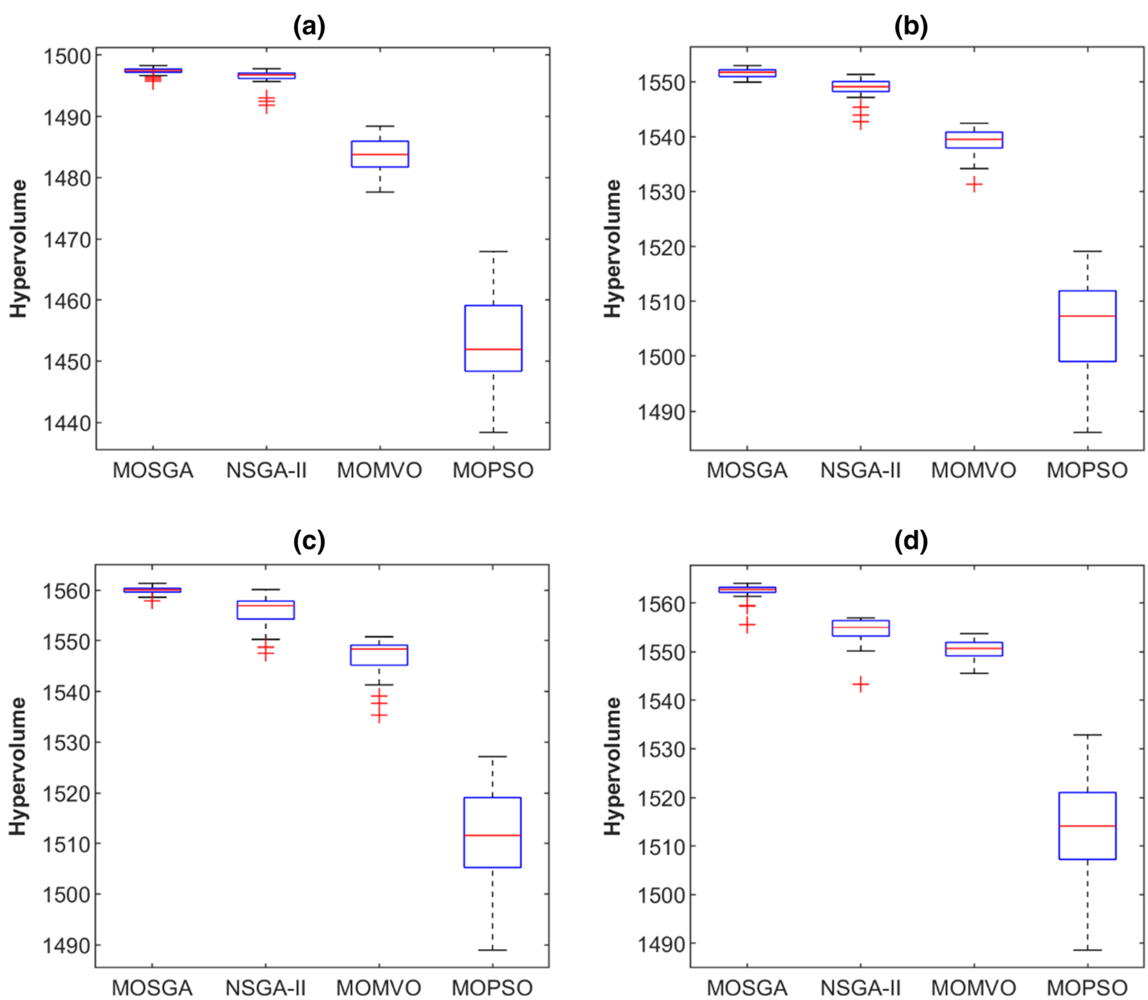


Fig. 13 Box plots of HV metric for four case studies: **a** pure water, **b** SiO_2, **c** Al_2O_3, **d** CuO

Table 20 Wilcoxon rank-sum test results based on the HV metric for the TEO-FPSC problem

MOSGA versus	NSGA-II		MOMVO		MOPSO	
	p value	Signed	*p* value	Signed	*p* value	Signed
Pure water	1.87E−05	+	3.02E−11	+	3.02E−11	+
SiO_2	5.07E−10	+	3.02E−11	+	3.02E−11	+
Al_2O_3	8.10E−10	+	3.02E−11	+	3.02E−11	+
CuO	1.21E−10	+	3.02E−11	+	3.02E−11	+

MOSGA obtained better performance than other algorithms in terms of convergence and distribution of Pareto optimal solutions.

5 Conclusion

This paper introduced the new multi-objective version of the SGA called MOSGA for the TEO-FPSC problem. Its key mechanism was inspired by the conventional SGA, in which non-dominated solutions were found through the stages of mutation, generation, and selection. Elitist non-dominated sorting technique and Pareto archive with selection instruments were integrated with SGA to produce MOSGA. To verify its efficacy, MOSGA was tested with eight multi-objective benchmark problems. In all examined cases, statistical results demonstrated that MOSGA effectively converged toward true Pareto optimal fronts with high distribution and spread for all generated fronts. After that, four case studies of FPSC systems with different working fluids (pure water, SiO_2, Al_2O_3, CuO) were optimized by using the MOSGA. It was found that different nanofluids enhanced both efficiency and TAC of FPSC systems. Results were also analyzed and compared with other multi-objective techniques in terms of three performance criteria: C-metric, SP, and HV metrics. Through all trials, MOSGA provided superior solutions compared to three other well-known multi-objective algorithms. This can be clearly seen through the CuO case study, where MOSGA solutions dominated more than 21.55% of NSGA-II solutions; 28.01% of MOMVO solutions; and 74.93% of MOPSO solutions. Pareto optimal fronts generated by MOSGA offered an essential approach to assist manufacturers in their determination of optimal trade-offs between thermal efficiency and TAC for FPSC systems when confronted with multi-objective problems. For future works, it is recommended to implement MOSGA for multi-objective optimization in other solar thermal systems such as parabolic trough collector, concentrated solar power, and solar power tower.

Acknowledgements This research is sponsored by YUTP–FRG and Graduate Research Assistantship (GRA) Scheme of Universiti Teknologi PETRONAS.

Declarations

Conflict of interest Our paper has no conflicts of interest with any other individuals or parties.

References

1. Agency IE (2018) Renewables 2018
2. Raj P, Subudhi S (2018) A review of studies using nanofluids in flat-plate and direct absorption solar collectors. Renew Sustain Energy Rev 84:54–74. https://doi.org/10.1016/j.rser.2017.10.012
3. Evangelisti L, De Lieto VR, Asdrubali F (2019) Latest advances on solar thermal collectors: a comprehensive review. Renew Sustain Energy Rev 114:109318. https://doi.org/10.1016/j.rser.2019.109318
4. (2018) Renewables 2018 global status report. In: Green growth knowledge platform. https://www.greengrowthknowledge.org/resource/renewables-2018-global-status-report. Accessed 31 Oct 2019
5. Kalogirou S (2003) The potential of solar industrial process heat applications. Appl Energy 76:337–361. https://doi.org/10.1016/S0306-2619(02)00176-9
6. Karki S, Haapala KR, Fronk BM (2019) Technical and economic feasibility of solar flat-plate collector thermal energy systems for small and medium manufacturers. Appl Energy 254:113649. https://doi.org/10.1016/j.apenergy.2019.113649
7. Jing OL, Bashir MJK, Kao J-J (2015) Solar radiation based benefit and cost evaluation for solar water heater expansion in Malaysia. Renew Sustain Energy Rev 48:328–335. https://doi.org/10.1016/j.rser.2015.04.031
8. Huy TH, Nallagownden P, Kannan R (2019) Energetic optimization of solar water heating system with flat plate collector using search group algorithm. J Adv Res Fluid Mech Therm Sci 61:17
9. Farahat S, Sarhaddi F, Ajam H (2009) Exergetic optimization of flat plate solar collectors. Renew Energy 34:1169–1174. https://doi.org/10.1016/j.renene.2008.06.014
10. Jafarkazemi F, Ahmadifard E (2013) Energetic and exergetic evaluation of flat plate solar collectors. Renew Energy 56:55–63. https://doi.org/10.1016/j.renene.2012.10.031
11. Badr O, Mohammed A, Brahim D (2018) Optimization of the thermal performance of the solar water heater (SWH) using stochastic technique 10
12. Wenceslas KY, Ghislain T (2019) Experimental validation of exergy optimization of a flat-plate solar collector in a thermosyphon solar water heater. Arab J Sci Eng 44:2535–2549. https://doi.org/10.1007/s13369-018-3227-x
13. Khademi M, Jafarkazemi F, Ahmadifard E, Younesnejad S (2012) Optimizing exergy efficiency of flat plate solar collectors using SQP and genetic algorithm. AMM 253–255:760–765. https://doi.org/10.4028/www.scientific.net/AMM.253-255.760

14. Siddhartha SN, Varun, (2012) A particle swarm optimization algorithm for optimization of thermal performance of a smooth flat plate solar air heater. Energy 38:406–413. https://doi.org/10.1016/j.energy.2011.11.026
15. Siddhartha, Chauhan SR, Varun, Sharma N (2011) Thermal performance optimization of smooth flat plate solar air heater (SFPSAH) using simulated annealing: evaluation and comparisons. In: 2011 international conference utility exhibition on power and energy systems: issues and prospects for Asia (ICUE), pp 1–5
16. Varun S (2010) Thermal performance optimization of a flat plate solar air heater using genetic algorithm. Appl Energy 87:1793–1799. https://doi.org/10.1016/j.apenergy.2009.10.015
17. Varun SN, Bhat IK, Grover D (2011) Optimization of a smooth flat plate solar air heater using stochastic iterative perturbation technique. Sol Energy 85:2331–2337. https://doi.org/10.1016/j.solener.2011.06.022
18. Rao RV, Waghmare G (2015) Optimization of thermal performance of a smooth flat-plate solar air heater using teaching–learning-based optimization algorithm. Cog Eng 2:997421. https://doi.org/10.1080/23311916.2014.997421
19. Şencan Şahin A (2012) Optimization of solar air collector using genetic algorithm and artificial bee colony algorithm. Heat Mass Transf 48:1921–1928. https://doi.org/10.1007/s00231-012-1038-2
20. Yıldırım C, Aydoğdu İ (2017) Artificial bee colony algorithm for thermohydraulic optimization of flat plate solar air heaters. J Mech Sci Technol 31:3593–3602. https://doi.org/10.1007/s12206-017-0647-6
21. Jiandong Z, Hanzhong T, Susu C (2015) Numerical simulation for structural parameters of flat-plate solar collector. Sol Energy 117:192–202. https://doi.org/10.1016/j.solener.2015.04.027
22. Bornatico R, Pfeiffer M, Witzig A, Guzzella L (2012) Optimal sizing of a solar thermal building installation using particle swarm optimization. Energy 41:31–37. https://doi.org/10.1016/j.energy.2011.05.026
23. Araya R, Bustos F, Contreras J, Fuentes A (2017) Life-cycle savings for a flat-plate solar water collector plant in Chile. Renew Energy 112:365–377. https://doi.org/10.1016/j.renene.2017.05.036
24. Ko MJ (2015) Analysis and optimization design of a solar water heating system based on life cycle cost using a genetic algorithm. Energies 8:11380–11403. https://doi.org/10.3390/en81011380
25. Cheng Hin JN, Zmeureanu R (2014) Optimization of a residential solar combisystem for minimum life cycle cost, energy use and exergy destroyed. Sol Energy 100:102–113. https://doi.org/10.1016/j.solener.2013.12.001
26. Yaman K, Arslan G (2018) Modeling, simulation, and optimization of a solar water heating system in different climate regions. J Renew Sustain Energy 10:023703. https://doi.org/10.1063/1.5004069
27. Kusyy O, Kuethe S, Vajen K (2010) Simulation-based optimization of a solar water heating system by a hybrid genetic-binary search algorithm. In: 2010 XVth international seminar/workshop on direct and inverse problems of electromagnetic and acoustic wave theory (DIPED), pp 201–206
28. Hajabdollahi Z, Hajabdollahi H (2017) Thermo-economic modeling and multi-objective optimization of solar water heater using flat plate collectors. Sol Energy 155:191–202. https://doi.org/10.1016/j.solener.2017.06.023
29. Hajabdollahi F, Premnath K (2017) Numerical study of the effect of nanoparticles on thermoeconomic improvement of a solar flat plate collector. Appl Therm Eng 127:390–401. https://doi.org/10.1016/j.applthermaleng.2017.08.058
30. Hajabdollahi H (2018) Investigating the effect of nanofluid on optimal design of solar flat plate collector. In: 2018 5th international conference on renewable energy: generation and applications (ICREGA), pp 188–191
31. Hajabdollahi Z, Hajabdollahi H, Kim KC (2019) Multi-objective optimization of solar collector using water-based nanofluids with different types of nanoparticles. J Therm Anal Calorim. https://doi.org/10.1007/s10973-019-08444-w
32. Gonçalves MS, Lopez RH, Miguel LFF (2015) Search group algorithm: a new metaheuristic method for the optimization of truss structures. Comput Struct 153:165–184. https://doi.org/10.1016/j.compstruc.2015.03.003
33. Pedro RL, Demarche J, Miguel LFF, Lopez RH (2017) An efficient approach for the optimization of simply supported steel-concrete composite I-girder bridges. Adv Eng Softw 112:31–45. https://doi.org/10.1016/j.advengsoft.2017.06.009
34. Carraro F, Lopez RH, Miguel LFF (2017) Optimum design of planar steel frames using the search group algorithm. J Braz Soc Mech Sci Eng 39:1405–1418. https://doi.org/10.1007/s40430-016-0628-1
35. Noorbin SFH, Alfi A (2018) Adaptive parameter control of search group algorithm using fuzzy logic applied to networked control systems. Soft Comput 22:7939–7960. https://doi.org/10.1007/s00500-017-2742-0
36. Khamari D, Sahu RK, Panda S (2019) Application of search group algorithm for automatic generation control of interconnected power system. In: Behera HS, Nayak J, Naik B, Abraham A (eds) Computational intelligence in data mining. Springer, Singapore, pp 557–568
37. Khamari D, Sahu RK, Panda S (2019) Application of search group algorithm for automatic generation control of multi-area multi-source power systems. E3S Web Conf 87:01005. https://doi.org/10.1051/e3sconf/20198701005
38. Acampora G, Caruso D, Vaccaro A, Vitiello A (2016) A search group algorithm for optimal voltage regulation in power systems. In: 2016 IEEE congress on evolutionary computation (CEC), pp 3662–3669
39. Sadollah A, Eskandar H, Bahreininejad A, Kim JH (2015) Water cycle algorithm for solving multi-objective optimization problems. Soft Comput 19:2587–2603. https://doi.org/10.1007/s00500-014-1424-4
40. Sadollah A, Eskandar H, Kim JH (2015) Water cycle algorithm for solving constrained multi-objective optimization problems. Appl Soft Comput 27:279–298. https://doi.org/10.1016/j.asoc.2014.10.042
41. Mirjalili S, Saremi S, Mirjalili SM, dos Coelho LS (2016) Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. Expert Syst Appl 47:106–119. https://doi.org/10.1016/j.eswa.2015.10.039
42. Tran D-H, Cheng M-Y, Prayogo D (2016) A novel multiple objective symbiotic organisms search (MOSOS) for time–cost–labor utilization tradeoff problem. Knowl Based Syst 94:132–145. https://doi.org/10.1016/j.knosys.2015.11.016
43. Mirjalili S, Jangir P, Mirjalili SZ et al (2017) Optimization of problems with multiple objectives using the multi-verse optimization algorithm. Knowl Based Syst 134:50–71. https://doi.org/10.1016/j.knosys.2017.07.018
44. Foroughi Nematollahi A, Rahiminejad A, Vahidi B (2019) A novel multi-objective optimization algorithm based on lightning attachment procedure optimization algorithm. Appl Soft Comput 75:404–427. https://doi.org/10.1016/j.asoc.2018.11.032
45. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1:67–82. https://doi.org/10.1109/4235.585893
46. Kalogirou SA (2004) Solar thermal collectors and applications. Prog Energy Combust Sci 30:231–295. https://doi.org/10.1016/j.pecs.2004.02.001

47. Agarwal VK, Larson DC (1981) Calculation of the top loss coefficient of a flat-plate collector. Sol Energy 27:69–71. https://doi.org/10.1016/0038-092X(81)90022-0
48. Mahian O, Kianifar A, Sahin AZ, Wongwises S (2015) Heat transfer, pressure drop, and entropy generation in a solar collector using SiO_2/water nanofluids: effects of nanoparticle size and pH. J Heat Transf. https://doi.org/10.1115/1.4029870
49. Pak BC, Cho YI (1998) Hydrodynamic and heat transfer study of dispersed fluids with submicron metallic oxide particles. Exp Heat Transf 11:151–170. https://doi.org/10.1080/08916159808946559
50. Xuan Y, Roetzel W (2000) Conceptions for heat transfer correlation of nanofluids. Int J Heat Mass Transf 43:3701–3707. https://doi.org/10.1016/S0017-9310(99)00369-5
51. Xuan Y, Li Q, Hu W (2003) Aggregation structure and thermal conductivity of nanofluids. AIChE J 49:1038–1043. https://doi.org/10.1002/aic.690490420
52. Maïga SEB, Nguyen CT, Galanis N, Roy G (2004) Heat transfer behaviours of nanofluids in a uniformly heated tube. Superlattices Microstruct 35:543–557. https://doi.org/10.1016/j.spmi.2003.09.012
53. Taal M, Bulatov I, Klemeš J, Stehlík P (2003) Cost estimation and energy price forecasts for economic evaluation of retrofit projects. Appl Therm Eng 23:1819–1835. https://doi.org/10.1016/S1359-4311(03)00136-4
54. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6:182–197. https://doi.org/10.1109/4235.996017
55. Goldberg DE (1989) Genetic algorithms in search, optimization and machine learning, 1st edn. Addison-Wesley Longman Publishing Co., Inc., Boston
56. Coello CAC, Lamont GB, Veldhuizen DAV (2006) Evolutionary algorithms for solving multi-objective problems (genetic and evolutionary computation). Springer, Berlin, Heidelberg
57. Veldhuizen DAV, Lamont GB (1998) Evolutionary computation and convergence to a Pareto front. Late breaking papers at the genetic programming 1998 conference. Stanford University Bookstore, Stanford, pp 221–228
58. Schott JR (1995) Fault tolerant design using single and multicriteria genetic algorithm optimization. Thesis, Massachusetts Institute of Technology
59. Zitzler E (1999) Evolutionary algorithms for multiobjective optimization: methods and applications. Swiss Federal Institute of Technology, Zurich
60. Deb K, Kalyanmoy D (2001) Multi-objective optimization using evolutionary algorithms. Wiley, New York
61. Zitzler E, Thiele L (1998) Multiobjective optimization using evolutionary algorithms—a comparative case study. In: Eiben AE, Bäck T, Schoenauer M, Schwefel H-P (eds) Parallel problem solving from nature—PPSN V. Springer, Berlin, pp 292–301
62. Tran D-H, Luong-Duc L, Duong M-T et al (2018) Opposition multiple objective symbiotic organisms search (OMOSOS) for time, cost, quality and work continuity tradeoff in repetitive projects. J Comput Des Eng 5:160–172. https://doi.org/10.1016/j.jcde.2017.11.008
63. Sheng W, Liu K-Y, Liu Y et al (2015) Optimal placement and sizing of distributed generation via an improved nondominated sorting genetic algorithm II. IEEE Trans Power Deliv 30:569–578. https://doi.org/10.1109/TPWRD.2014.2325938
64. Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput 8:173–195. https://doi.org/10.1162/106365600568202
65. Schaffer JD (1984) Some experiments in machine learning using vector evaluated genetic algorithms (artificial intelligence, optimization, adaptation, pattern recognition). Phd, Vanderbilt University
66. Fonseca CM, Fleming PJ (1995) An overview of evolutionary algorithms in multiobjective optimization. Evol Comput 3:1–16. https://doi.org/10.1162/evco.1995.3.1.1
67. Poloni C, Giurgevich A, Onesti L, Pediroda V (2000) Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics. Comput Methods Appl Mech Eng 186:403–420. https://doi.org/10.1016/S0045-7825(99)00394-1
68. Kursawe F (1991) A variant of evolution strategies for vector optimization. In: Schwefel H-P, Männer R (eds) Parallel problem solving from nature. Springer, Berlin, Heidelberg, pp 193–197
69. Coello CAC, Pulido GT, Lechuga MS (2004) Handling multiple objectives with particle swarm optimization. IEEE Trans Evol Comput 8:256–279. https://doi.org/10.1109/TEVC.2004.826067
70. Flat plate panels. In: Kingspan|USA. https://www.kingspan.com/us/en-us/product-groups/renewable-technologies/solar-thermal/solar-flat-plate-panels/flat-plate-panels. Accessed 6 Nov 2019